**Test 2**

**Tanner Lang**

**UID: 815357782**

**Test Details:**

Test1: (20 points)  
Steps to Perform:  
1. Navigate to the URL: https://demoblaze.com/  
2. Access the Home Page.  
3. Scroll down using any locator of your choice, click on the "Samsung Galaxy S6" link,  
and add the item to the cart.  
4. Display the confirmation message upon successful addition.  
5. Retrieve and print the current URL and page title in the output console.  
6. Specify the size of the page and list all web elements in the console.

The images are shown below.  
===================================================================  
======================================================================

Test 2: (20 points)  
Write a Selenium WebDriver test script to automate the following scenarios:  
Website for Testing:  
Demo Site: https://www.saucedemo.com/  
• Username: standard\_user  
• Password: secret\_sauce  
• Wrong Credentials: Any invalid username/password  
1.Successful Login:  
o Navigate to the website's login page.  
o Enter a valid email and password.  
o Click the Login button.  
o Verify that the user is redirected to the dashboard and their name is displayed.  
2. Unsuccessful Login:  
• Enter an incorrect email or password.  
• Click the Login button.  
• Verify that an error message appears.  
Image of the website: https://www.saucedemo.com/

Test 3 (20 points):  
Steps to Perform:  
1. Navigate to the URL:  
https://www.tutorialspoint.com/selenium/practice/selenium\_automation\_practice.p  
hp.  
2. Use the link text locator to click on the "Link" button, then select and click on the first  
"Home" link.  
3. Retrieve and print the current URL of the webpage.  
4. Capture a screenshot of the Home page (link text) and save it to a specified file  
location.

Test 4: (20 points)  
Steps to Perform:  
1. Navigate to the URL:  
https://www.tutorialspoint.com/selenium/practice/selenium\_automation\_practice.p  
hp.  
2. Use any suitable locator to click on the "Upload and Download" button.  
3. Upload an image file by specifying the file path.  
4. After uploading, verify that the image file has been successfully uploaded and is visible on  
the page.

Test 5: (20 points)  
Steps to Perform:  
1. Navigate to the URL: https://jqueryui.com/sortable/.  
2. Use any locator of your choice to navigate to the "Browser" tab.  
3. Select "Draggable" and perform the drag-and-drop action.  
4. Choose "Resizable" and execute the resizing function.  
5. Navigate to "Selectable" and perform the selection action

**CODE**

Main.java

import java.io.IOException;  
  
public class Main {  
 public static void main(String[] args) throws InterruptedException, IOException {  
 TestCases test = new TestCases();  
  
  
 //test 1  
 test.driverSetup();  
 test.Test1();  
 test.cleanUp();  
  
 //test 2  
 test.driverSetup();  
 test.Test2();  
 test.cleanUp();  
  
 //test 3  
 test.driverSetup();  
 test.test3();  
 test.cleanUp();  
  
 //test 4  
 test.driverSetup();  
 test.Test4();  
 test.cleanUp();  
  
 //test 5  
 test.driverSetup();  
 test.Test5();  
 test.cleanUp();  
 }  
}

Next Page:

TestCases.java

// click the <icon src="AllIcons.Actions.Execute"/> icon in the gutter.  
import org.openqa.selenium.\*;  
import org.openqa.selenium.chrome.ChromeDriver;  
import org.openqa.selenium.interactions.Actions;  
import org.openqa.selenium.io.FileHandler;  
import org.testng.Assert;  
import org.testng.annotations.\*;  
import org.openqa.selenium.JavascriptExecutor;  
import org.testng.thread.IThreadWorkerFactory;  
  
import java.io.File;  
import java.io.IOException;  
import java.util.ArrayList;  
import java.util.List;  
import java.util.Set;  
import java.util.concurrent.TimeUnit;  
  
// click the <icon src="AllIcons.Actions.Execute"/> icon in the gutter.  
  
public class TestCases  
  
{  
 WebDriver driver; //init WebDriver  
  
  
 //Call to setup the chrome driver for testing.  
 @BeforeMethod  
 public void driverSetup()  
 {  
 System.*setProperty*("webdriver.chrom.driver","C://Users//tjlan//OneDrive//Desktop//Software Testing//Drivers//chromedriver.exe");  
 driver = new ChromeDriver();  
 driver.manage().window().maximize();  
 }  
 //call after all tests have completed, exits the test browser and closes the test.  
 @AfterMethod  
 public void cleanUp()  
 {  
 //wait for half a second  
 try {  
 Thread.*sleep*(500);  
 } catch (InterruptedException e) {  
 throw new RuntimeException(e);  
 }  
 //close the test, if driver is open  
 if(driver != null)  
 {  
 driver.quit();  
 }  
 }  
  
  
 @Test(priority = 1)  
 public void Test1() throws InterruptedException {  
 System.*out*.println("-------------------TEST 1----------------------");  
 //Navigates to demoblaze  
 driver.get("https://demoblaze.com/");  
  
 //create javascript executor for scrolling  
 JavascriptExecutor exe = (JavascriptExecutor)driver;  
  
 //wait for 4 seconds and scroll  
 Thread.*sleep*(2000);  
 exe.executeScript("window.scroll(0,300)", ""); //scrolls 300 pixels down  
  
  
 //selects the Samsung Galaxy S6 link via cssSelector, which takes us to a new page.  
 Thread.*sleep*(2000);  
 driver.findElement(By.*cssSelector*("a[href='prod.html?idp\_=1']")).click(); //finds by css, clicks  
  
 //wait a couple seconds and add the s6 to the cart, via finding the element by cssSelector.  
 Thread.*sleep*(2000);  
 driver.findElement(By.*cssSelector*("a.btn-success[onclick\*='addToCart']")).click(); //finds by css, clicks  
 Thread.*sleep*(500);  
  
 // Try to handle the alert without blocking the test  
 try {  
 Alert alert = driver.switchTo().alert();  
 System.*out*.println("Alert found: " + alert.getText());  
 alert.accept();  
 } catch (Exception e) {  
 System.*out*.println("No alert found, continuing execution.");  
 }  
  
 //Get the current url and page title  
 String url = driver.getCurrentUrl(); //gets url, stores in string url  
 String title = driver.getTitle(); //getst title, stores in string title  
 System.*out*.println("Title: " + title); //prints  
 System.*out*.println("URL: " + url); //prints  
  
 //Specify Size of Page and print to console  
 int width = driver.manage().window().getSize().getWidth(); //gets width  
 int height = driver.manage().window().getSize().getHeight(); //gets height  
 System.*out*.println("Width: " + width + " Height: " + height); //print statements  
  
  
  
 //List all elements into the console -------  
 List<WebElement> pageElements = driver.findElements(By.*xpath*("//\*")); //add all elements to list  
 System.*out*.println("Total elements found: " + pageElements.size()); //list total elements found  
  
 //iterate through list containing elements, print every element in the list.  
 for (WebElement element : pageElements) {  
 try {  
 System.*out*.println("Tag: " + element.getTagName());  
 System.*out*.println("Text: " + element.getText());  
 System.*out*.println("Attributes:");  
 System.*out*.println(" id: " + element.getAttribute("id"));  
 System.*out*.println(" class: " + element.getAttribute("class"));  
 System.*out*.println(" src: " + element.getAttribute("src"));  
 System.*out*.println(" href: " + element.getAttribute("href"));  
 System.*out*.println("--------------------------------");  
 } catch (Exception e) {  
 System.*out*.println("Error retrieving element details.");  
 }  
 }  
  
 //wait before moving on to Test2  
 Thread.*sleep*(1500);  
  
 }  
  
  
  
 @Test(priority = 2)  
 public void Test2() throws InterruptedException {  
  
 System.*out*.println("-------------------TEST 2----------------------");  
 driver.get("https://www.saucedemo.com/");  
 Thread.*sleep*(2000); //wait for a couple seconds before entering login credentials.  
  
 // Correct Login  
 driver.findElement(By.*id*("user-name")).sendKeys("standard\_user"); //enter usernmae  
 driver.findElement(By.*id*("password")).sendKeys("secret\_sauce"); //enter password  
 Thread.*sleep*(3000); //wait for a few seconds before clicking login.  
 driver.findElement(By.*id*("login-button")).click(); //find login button and click  
  
 //check success and confirm in console  
 String expectedURL = "https://www.saucedemo.com/inventory.html";  
 //assert checks if urls are equal, if they are test continues.  
 Assert.*assertEquals*(driver.getCurrentUrl(), expectedURL, "Login failed!");  
  
 String trueURL = driver.getCurrentUrl();  
 if (expectedURL.equals(trueURL))  
 {  
 System.*out*.println("Correct Login Test: Login Successful, Test Passed");  
 }  
 else  
 {  
 System.*out*.println("Login Failed");  
 }  
  
 //verify title element is displayed  
 WebElement inventoryTitle = driver.findElement(By.*className*("title"));  
 //test continues if true, otherwise it fails and prints:  
 Assert.*assertTrue*(inventoryTitle.isDisplayed(), "Inventory page is not displayed");  
 //wait before testing the incorrect login  
 Thread.*sleep*(5000);  
  
 //Navigate back to the login page.  
 driver.get("https://www.saucedemo.com/");  
 Thread.*sleep*(2000); //wait for a couple seconds before entering the incorrecgt login credentials.  
 driver.findElement(By.*id*("user-name")).sendKeys("wrong"); //enter incorrect usernmae  
 driver.findElement(By.*id*("password")).sendKeys("wrong"); //enter password  
 Thread.*sleep*(3000); //wait for a few seconds before clicking login.  
 driver.findElement(By.*id*("login-button")).click(); //find login button and click  
  
 //Confirm error message appears, print to console that incorrect login test passed  
 //inits webelement errorMessage to be the error message on the page.  
 WebElement errorMessage = driver.findElement(By.*cssSelector*("[data-test='error']"));  
 if(errorMessage.isDisplayed())  
 {  
 System.*out*.println("Incorrect Login Test: Login Unsuccessful, Test Passed");  
 }  
 //if true, test continues, otherwise test failes and displays:  
 Assert.*assertTrue*(errorMessage.isDisplayed(), "Error message not shown for invalid login");  
 //if equal, test continues, otherwise it fails.  
 Assert.*assertEquals*(errorMessage.getText(), "Epic sadface: Username and password do not match any user in this service");  
  
  
 //wait before moving on to Test3  
 Thread.*sleep*(4000);  
  
 }  
  
 @Test(priority = 3)  
 public void test3() throws InterruptedException, IOException {  
 System.*out*.println("-------------------TEST 3----------------------");  
  
 //navigate to test 3 site  
 driver.get("https://www.tutorialspoint.com/selenium/practice/selenium\_automation\_practice.php");  
 Thread.*sleep*(2000); //let the webpage load.  
  
 //open up the elements dropdown, locate via xpath  
 WebElement dropdownMenu = driver.findElement(By.*xpath*("//button[contains(@class, 'accordion-button')]"));  
 dropdownMenu.click(); // Expand the dropdown  
 Thread.*sleep*(2000); //wait a few seconds after opening the dropdown.  
  
  
 //use the linktext locator to click on link  
 WebElement link = driver.findElement(By.*linkText*("Links"));  
 link.click();  
 Thread.*sleep*(2000); //wait a couple seconds before clicking on home  
  
  
 //use the linktext locator to click on the home link  
 WebElement home = driver.findElement(By.*linkText*("Home"));  
 home.click();  
  
  
 Thread.*sleep*(2000); //after clicking home, wait before switching tabs.  
  
 //webhandle to switch tabs to the new tab  
 ArrayList<String> tabs = new ArrayList<String> (driver.getWindowHandles());  
 driver.switchTo().window(tabs.get(1));  
  
 //get the current url  
 String url = driver.getCurrentUrl();  
 System.*out*.println("Current URL: " + url);  
  
 //take a screenshot.  
 try {  
 TakesScreenshot screenshot = (TakesScreenshot) driver;  
 File src = screenshot.getScreenshotAs(OutputType.*FILE*);  
 //saves screenshot to folder  
 File des = new File("C://Users//tjlan//OneDrive//Desktop//Software Testing//ST\_Test2\_AutomationTesting//Screenshot//screenshot.png");  
 FileHandler.*copy*(src, des);  
  
 }catch (IOException e){  
 System.*out*.println("Screenshot failed.");  
 }  
  
  
 Thread.*sleep*(3000); //wait a couple of seconds on the home page before the test ends.  
  
 }  
  
  
 @Test(priority = 4)  
 public void Test4() throws InterruptedException {  
 System.*out*.println("-------------------TEST 4----------------------");  
  
 //navigate to test 3 site  
 driver.get("https://www.tutorialspoint.com/selenium/practice/selenium\_automation\_practice.php");  
 Thread.*sleep*(2000); //let the webpage load.  
  
 //open up the elements dropdown, locate via xpath  
 WebElement dropdownMenu = driver.findElement(By.*xpath*("//button[contains(@class, 'accordion-button')]"));  
 dropdownMenu.click(); // Expand the dropdown  
 Thread.*sleep*(2000); //wait a few seconds after opening the dropdown.  
  
 //use an xpath locator to click on the upload and download button  
 WebElement upDownButton = driver.findElement(By.*xpath*("//a[@href='upload-download.php']"));  
 upDownButton.click(); //click on the up down button  
  
 Thread.*sleep*(1500); //wait a little bit before we take a screenshot.  
 //upload the screenshot we took in test 3.  
 WebElement uploadInput = driver.findElement(By.*id*("uploadFile"));  
 uploadInput.sendKeys("C://Users//tjlan//OneDrive//Desktop//Software Testing//ST\_Test2\_AutomationTesting//Screenshot//screenshot.png");  
  
 //confirms upload  
 String uploadedFileName = uploadInput.getAttribute("value");  
 System.*out*.println("Uploaded file: " + uploadedFileName);  
  
 // Assertion, test continues if file uploaded, otherwise test fails.  
 Assert.*assertTrue*(uploadedFileName.contains("screenshot.png"), "File upload failed!");  
  
 System.*out*.println("File uploaded successfully!");  
  
 Thread.*sleep*(2000); // wait  
  
 }  
  
  
 @Test(priority = 5)  
 public void Test5() throws InterruptedException  
 {  
  
 JavascriptExecutor exe = (JavascriptExecutor)driver;  
  
 System.*out*.println("-------------------TEST 5----------------------");  
 driver.get("https://jqueryui.com");  
 Thread.*sleep*(1000);  
  
 //using the xpath locator to click on the draggable tab  
 WebElement draggableButton = driver.findElement(By.*xpath*("//a[text()='Draggable']"));  
 draggableButton.click();  
 Thread.*sleep*(2000); //wait two seconds after clicking  
  
 //drag the thing around  
 exe.executeScript("window.scroll(0,300)", ""); //scrolls 300 pixels down  
 Thread.*sleep*(500); //briefly wait after scrolling  
 driver.switchTo().frame(0); //switch to iframe  
 WebElement draggable = driver.findElement(By.*id*("draggable"));  
 Actions drag = new Actions(driver); //create a new action called drag  
 drag.dragAndDropBy(draggable, 100, 100).perform(); //perform the action  
 Thread.*sleep*(1000); //wait for 1 second before moving on  
 driver.switchTo().defaultContent(); //switch off the iframe to move on  
  
 //use an xpath locator to click on the resizeable tab.  
 WebElement resizeableButton = driver.findElement(By.*xpath*("//a[text()='Resizable']"));  
 resizeableButton.click();  
 Thread.*sleep*(2000); //wait two seconds after clicking  
  
 exe.executeScript("window.scroll(0,300)", ""); //scrolls 300 pixels down  
 Thread.*sleep*(500); //briefly wait after scrolling  
 driver.switchTo().frame(0); //switch to iframe  
 WebElement resizable = driver.findElement(By.*cssSelector*(".ui-resizable-se"));  
 Actions resize = new Actions(driver); //create a new action called resize  
 resize.clickAndHold(resizable).moveByOffset(100, 100).release().perform();  
 Thread.*sleep*(1000); //wait for 1 second before moving on  
 driver.switchTo().defaultContent(); //switch off the iframe to move on  
  
 //use an xpath locator to click on the selector tab  
 WebElement selectableButton = driver.findElement(By.*xpath*("//a[text()='Selectable']"));  
 selectableButton.click();  
 Thread.*sleep*(2000); //wait two seconds before clicking  
  
 exe.executeScript("window.scroll(0,300)", ""); //scrolls 300 pixels down  
 Thread.*sleep*(500); //briefly wait after scrolling  
 driver.switchTo().frame(0); //switch to iframe  
 WebElement selectable = driver.findElement(By.*xpath*("//li[text()='Item 1']"));  
 Actions select = new Actions(driver); //creates a new action called select  
 select.click(selectable).perform(); //perform selecting the select item  
 driver.switchTo().defaultContent(); //switch off the iframe to move on  
  
 }  
  
  
  
  
  
}