



UNIVERSITY OF VOCATIONAL TECHNOLOGY

100, KANDAWALA ROAD, RATMALANA

**Batchlor of Education in Technology.**

## **ASSIGNMENT -03**

**SUBJECT: Software Testing and Reliability**

**L.G.N.S.Gamage**

**BED/19/B2/08**

**4<sup>th</sup> SEMESTER**

# Test Case: -TC/01

Module: <https://www.saucedemo.com>

Sprint: Sprint 1

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check weather web page open successfully.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/01
<b>Test Case</b>	Check weather web page open successfully.
<b>Test Steps</b>	1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL
<b>Input data</b>	N/A
<b>Expected Results</b>	<a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web site should open
<b>Actual Results</b>	Opened <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure:** The Package Explorer view shows a project named "test" containing a package "test" which has a file "dddd.java". Other folders like "src" and "Chrome Driver" are also visible.
- Code Editor:** The main editor window displays the Java code for "dddd.java". The code uses Selenium WebDriver to open the "saucedemo" website and verify its title.
- Outline View:** The Outline view on the right shows the class "dddd" and its main method.
- Console View:** The bottom console shows the execution output. It includes log messages from Selenium, the title of the page ("Swag Labs"), and a success message ("Verification Successful - The correct Url is opened").

```
package test;
import org.openqa.selenium.WebDriver;
public class dddd {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "C:\\Users\\Poornima Gamage\\eclipse-workspace\\test\\src\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        String baseURL="https://www.saucedemo.com/";
        driver.get(baseURL);
        String title = driver.getTitle();
        int titleLength = driver.getTitle().length();
        System.out.println("Title of the page is : " + title);
        // Storing URL in String variable
        String actualUrl = driver.getCurrentUrl();
        if (actualUrl.equals("https://www.saucedemo.com/")){
            System.out.println("Verification Successful - The correct Url is opened.");
        }
        else{
            System.out.println("Verification Failed - An incorrect Url is opened.");
        }
    }
}
```

```
<terminated> dddd [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.core_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 3, 2022)
INFO: Detected upstream dialect: W3C
Jul 03, 2022 12:09:57 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Title of the page is : Swag Labs
Verification Successful - The correct Url is opened.
```

## Test Case: -TC/02

Module: <https://www.saucedemo.com>

Sprint: Sprint 2

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Normal
<b>Test Case ID</b>	TC/03
<b>Test Case</b>	Check whether Swag Lab image displayed successfully.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Check swag Lab image.</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Swag Lab Image should displayed
<b>Actual Results</b>	Swag Lab image displayed successfully.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - test/src/test/imageCheck.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbar:** Standard Eclipse toolbar icons.
- Left Sidebar:** Package Explorer (test) and Outline (test package, imageCheck class, main method).
- Central Area:** Code editor showing Java code for a Selenium test case named imageCheck. The code initializes a ChromeDriver, navigates to the Saucedemo URL, finds the logo element using XPath, and prints "Logo present" if found or "Logo not present" if not.
- Right Sidebar:** Outline view showing the project structure and the main method.
- Bottom Area:** Problems, Javadoc, Declaration, Console tabs. The Console tab shows log output from the Java application, including Selenium startup logs and the printed message "Logo present".
- Taskbar:** Shows the taskbar with various pinned icons like File Explorer, Task View, and others.
- System Tray:** Shows system status icons including battery level (85%), network, and date/time (9:51 PM, 7/5/2022).

## Test Case: -TC/03

Module: <https://www.saucedemo.com>

Sprint: Sprint 2

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/03
<b>Test Case</b>	Check whether user able to login using only username.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter Only a correct username</li><li>3. Enter Submit button</li></ol>
<b>Input data</b>	Standard_user
<b>Expected Results</b>	Unable to login without Password
<b>Actual Results</b>	Unable to login without Password
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Project Structure:** The Package Explorer view shows a project named "test" containing a "src" folder with files: dddd.java, imageCheck.java, and LoginPageUser.java.
- Code Editor:** The main editor window displays the Java code for the LoginPageUser class. The code uses Selenium WebDriver to navigate to the saucedemo login page, enter a standard user, and click the login button. It then compares the actual URL with the expected URL to determine if the login was successful.
- Outline View:** The Outline view on the right shows the class structure, including the LoginPageUser class and its main method.
- Console View:** The bottom console shows the terminal output of the application running, indicating it failed to log in without a password.
- Taskbar:** The taskbar at the bottom includes the Windows Start button, a search bar, and various pinned application icons like File Explorer, Google Chrome, and Microsoft Word.

```
1 package test;
2 import java.util.concurrent.TimeUnit;
3 import org.openqa.selenium.By;
4 import org.openqa.selenium.WebDriver;
5 import org.openqa.selenium.WebElement;
6 import org.openqa.selenium.chrome.ChromeDriver;
7 import org.testng.Assert;
8
9
10 public class LoginPageUser {
11
12     public static void main(String[] args) {
13         System.setProperty("webdriver.chrome.driver", "C:\\\\Users\\\\Poornima Gamage\\\\eclipse-workspace\\\\test\\\\src\\\\test\\\\chromedriver.exe");
14         WebDriver driver=new ChromeDriver();
15         driver.manage().window().maximize();
16
17         driver.get("https://www.saucedemo.com/");
18
19         WebElement username=driver.findElement(By.id("user-name"));
20
21         WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
22         username.sendKeys("standard_user");
23         login.click(); String actualUrl="https://www.saucedemo.com/";
24         String expectedUrl= driver.getCurrentUrl();
25         if(actualUrl.equalsIgnoreCase(expectedUrl)) {
26             System.out.println("Unable to login without Password");
27         } else {
28             System.out.println("Able to login without Password");
29         }
30     }
31 }
```

```
<terminated> LoginPageUser [Java Application] C:\Users\Poornima Gamage\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 06, 2022 9:15:14 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 06, 2022 9:15:14 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Unable to login without Password
```

## Test Case: -TC/04

Module: <https://www.saucedemo.com>

Sprint: Sprint 2

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/04
<b>Test Case</b>	Check whether user able to login using only username.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter a only invalid username</li><li>3. Enter Submit button</li></ol>
<b>Input data</b>	My_username
<b>Expected Results</b>	Unable to login with incorrect username
<b>Actual Results</b>	Unable to login with incorrect username
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure (Package Explorer):** The project structure shows a package named "test" containing a source folder "src" with files: dddd.java, imageCheck.java, loginPageUser.java, and invalid\_userName.java.
- Code Editor (invalid\_userName.java):** The code implements a Selenium test for logging in with an invalid username. It imports necessary classes from org.openqa.selenium and org.openqa.selenium.chrome. The main method sets the chromedriver path, opens the saucedemo login page, finds the username field, sends the invalid username "My\_username", and clicks the login button. It then compares the actual URL with the expected URL ("https://www.saucedemo.com") to determine if the login was successful.
- Output (Console):** The terminal output shows the test execution results. It prints "Invalid User" because the actual URL did not match the expected URL.
- Bottom Bar:** The taskbar includes icons for search, file operations, and system status (temperature, battery, date).

```
package test;
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.junit.Assert;
public class invalid_userName {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "C:\\Users\\Poornima Gamage\\eclipse-workspace\\test\\src\\test\\Chrome Driver");
        WebDriver driver=new ChromeDriver();
        driver.manage().window().maximize();
        driver.get("https://www.saucedemo.com/");
        WebElement username=driver.findElement(By.id("user-name"));
        WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
        username.sendKeys("My_username");
        login.click(); String actualUrl="https://www.saucedemo.com/";
        String expectedUrl= driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
            System.out.println("Invalid User");
        } else {
            System.out.println("Valid User");
        }
    }
}
```

```
<terminated> invalid_userName [Java Application] C:\Users\Poornima Gamage.p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe
Jul 06, 2022 9:53:37 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 06, 2022 9:53:38 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Invalid User
```

## Test Case: -TC/05

Module: <https://www.saucedemo.com>

Sprint: Sprint 2

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/05
<b>Test Case</b>	Check whether user able to login using only incorrect password.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter a only invalid password</li><li>3. Enter Submit button</li></ol>
<b>Input data</b>	password
<b>Expected Results</b>	Unable to login with incorrect password
<b>Actual Results</b>	Unable to login with incorrect password
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar with various icons for file operations, search, and preferences.
- Package Explorer:** Shows the project structure with packages "src" and "test". Under "test", files "dddd.java", "imageCheck.java", "invalid\_userName.java", "loginPageUser.java", and "incorrect\_password.java" are listed.
- Editor:** The main editor window displays the Java code for the "incorrect\_password" class. The code uses Selenium WebDriver to navigate to the Saucedemo login page, enter an invalid password, and check if the login fails.

```
import java.util.concurrent.TimeUnit;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;

public class incorrect_password {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "C:\\Users\\Poornima Gamage\\eclipse-workspace\\test\\Chrome Driver\\chromedriver.exe");
        WebDriver driver=new ChromeDriver();
        driver.manage().window().maximize();

        driver.get("https://www.saucedemo.com/");

        WebElement password=driver.findElement(By.id("password"));

        WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
        password.sendKeys("password");
        login.click(); String actualUrl="https://www.saucedemo.com/";
        String expectedUrl= driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
            System.out.println("Incorrect Password");
        } else {
            System.out.println("Able to login with incorrect password");
        }
    }
}
```

- Outline View:** Shows the class structure with the main method.
- Problems View:** Displays log messages from the application's execution.

```
Jul 06, 2022 10:00:41 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 06, 2022 10:00:41 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Incorrect Password
```

- Console View:** Shows the standard Java console output.
- Taskbar:** Includes the Windows taskbar with the Eclipse icon, a search bar, pinned application icons (File Explorer, Mail, Task View, etc.), and system status indicators (battery, signal, volume).

## Test Case: -TC/06

Module: <https://www.saucedemo.com>

Sprint: Sprint 2

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/06
<b>Test Case</b>	Check whether user able to login using without username.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter a only password</li><li>3. Enter Submit button</li></ol>
<b>Input data</b>	secret_sauce
<b>Expected Results</b>	Unable to login without username
<b>Actual Results</b>	Unable to login with without username
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure:** The Package Explorer view shows a project named "test" containing a "src" folder with a "test" package. Inside "test", there are several Java files: "ddd.java", "imageCheck.java", "loginPageUser.java", "invalid\_userName.java", and "incorrect\_password.java".
- Code Editor:** The main editor window displays the "incorrect\_password.java" file. The code is a Java class named "incorrect\_password" with a main method. It uses Selenium WebDriver to navigate to "https://www.saucedemo.com/", find the password input field, send the text "secret\_sauce", click the login button, and then check if the actual URL matches the expected URL ("https://www.saucedemo.com/"). If it does, it prints "Login Sucess"; otherwise, it prints "Username is required".
- Outline View:** The Outline view on the right shows the class "incorrect\_password" and its main method.
- Console:** The bottom-right panel shows the terminal output of the application's execution. It includes log messages from Selenium and the printed output "Username is required".
- Taskbar:** The bottom of the screen shows the Windows taskbar with various pinned icons and the system tray.

## Test Case: -TC/07

Module: <https://www.saucedemo.com>

Sprint: Sprint 3

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/07
<b>Test Case</b>	Check whether user able to login using incorrect password & username
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter a only invalid username &amp; password</li><li>3. Enter Login button</li></ol>
<b>Input data</b>	user,password
<b>Expected Results</b>	Unable to login with incorrect username & password
<b>Actual Results</b>	Unable to login with incorrect username & password
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure:** The Package Explorer view shows a project named "test" containing a "src" folder with several Java files: dddd.java, imageCheck.java, incorrect\_password.java, incorrect\_UserPass.java, invalid\_userName.java, and LoginPageUser.java. It also includes "Chrome Driver" and "JRE System Library [JavaSE-17]" entries.
- Code Editor:** The main editor window displays the Java code for the "incorrect\_UserPass" class. The code uses Selenium WebDriver to interact with the Saucedemo login page, specifically targeting the user name and password fields and the login button. It checks if the actual URL after clicking the button matches the expected URL.
- Outline View:** The Outline view on the right side shows the package structure and the class "incorrect\_UserPass" with its single method "main".
- Console:** The bottom console window shows the terminal output of the Java application, indicating the execution of the test case and the resulting log messages.
- Taskbar:** The taskbar at the bottom of the screen shows various open applications including a search bar, file explorer, browser, and other productivity tools.

## Test Case: -TC/07

Module: <https://www.saucedemo.com>

Sprint: Sprint 3

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/07
<b>Test Case</b>	Check whether user able to login using valid username and password
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Enter a only username and password</li><li>3. Enter Submit button</li></ol>
<b>Input data</b>	standard_user / secret_sauce
<b>Expected Results</b>	User should able to login with correct username and password
<b>Actual Results</b>	Able to login with correct username and password
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar icons.
- Package Explorer:** Shows a project named "test" containing a package "src" with files: correct\_Logins.java, dddd.java, imageCheck.java, incorrect\_password.java, incorrect\_UserPass.java, invalid\_userName.java, LoginPageUser.java, and Chrome Driver.
- Outline View:** Shows the class structure: test > correct\_Logins > main(String[] args).
- Code Editor:** Displays the Java code for the `correct_Logins` class:

```
public class correct_Logins {
    public static void main(String[] args) {
        System.setProperty("webdriver.chrome.driver", "C:\\Users\\Poornima Gamage\\eclipse-workspace\\test\\src\\Chrome Driver\\chromedriver.exe");
        WebDriver driver=new ChromeDriver();
        driver.manage().window().maximize();

        driver.get("https://www.saucedemo.com/");

        WebElement username=driver.findElement(By.id("user-name"));
        WebElement password=driver.findElement(By.id("password"));

        WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
        username.sendKeys("standard_user");
        password.sendKeys("secret_sauce");
        login.click(); String actualUrl=driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
            System.out.println("Correct Login Details");
        } else {
            System.out.println("Login Success");
        }
    }
}
```
- Console:** Shows log output from the Java application:

```
<terminated> correct_Logins [Java Application] C:\Users\Poornima Gamage\.p2\pool\plugins\org.eclipse.jdt.core\org.eclipse.jdt.launching\ProtocolHandshake createSession
Jul 06, 2022 10:35:08 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 06, 2022 10:35:11 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Login Success
```
- Taskbar:** Shows the Windows taskbar with various pinned icons like File Explorer, Task View, Edge, Google Chrome, Mail, etc.

## Test Case: -TC/08

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/08
<b>Test Case</b>	Check whether inventory image displayed successfully.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Login the saucedemo web site using correct username and password</li><li>2. Check inventory images are visible correctly.</li></ol>
<b>Input data</b>	standard_user / secret_sauce
<b>Expected Results</b>	Image should displayed on inventory page.
<b>Actual Results</b>	Image displayed on inventory page.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar icons.
- Package Explorer:** Shows a project named "test" containing files like "inventory\_img.java", "correct\_Logins.java", and "imageCheck.java".
- Code Editor:** Displays the content of "inventory\_img.java". The code is a Java class named "inventory\_img" with a main method that sets up a ChromeDriver, inputs credentials, and checks for a specific logo image.
- Outline View:** Shows the class structure with a main method.
- Console:** Shows the output of the Java application, indicating successful login and the presence of the inventory image.
- System Tray:** Shows the Windows taskbar with various pinned icons and system status.

```
1 package test;
2
3 import org.openqa.selenium.By;
4
5 public class inventory_img {
6
7
8     public static void main(String[] args) {
9         System.setProperty("webdriver.chrome.driver", "C:\\\\Users\\\\Poornima Gamage\\\\eclipse-workspace\\\\test\\\\chromedriver.exe");
10        WebDriver driver=new ChromeDriver();
11        driver.manage().window().maximize();
12        driver.get("https://www.saucedemo.com/");
13
14        WebElement username=driver.findElement(By.id("user-name"));
15        WebElement password=driver.findElement(By.id("password"));
16
17        WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
18        username.sendKeys("standard_user");
19        password.sendKeys("secret_sauce");
20        login.click(); String actualUrl=driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
21            System.out.println("Incorrect Login Details");
22        } else {
23            System.out.println("Login Success");
24        }
25        WebElement logo_img = driver.findElement(By.xpath("//*[@id='item_4_img_link']/img"));
26    }
27
28 }
```

```
Jul 07, 2022 11:00:12 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 07, 2022 11:00:12 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Login Success
Inventory image present
```

## Test Case: -TC/09

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/09
<b>Test Case</b>	Check whether users able to click to side menu button.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password</li><li>3. Click side menu bar.</li></ol>
<b>Input data</b>	standard_user / secret_sauce
<b>Expected Results</b>	Side menu button should open with menu items
<b>Actual Results</b>	Side menu button opened with menu items
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows a project named "test" containing files "test\_navBar.java" and "inventory\_img.java".
- Code Editor:** Displays Java code for a Selenium test case. The code logs in to the saucedemo website, checks if the login was successful, and then performs actions on the side menu to verify its functionality.
- Outline View:** Shows the class structure with a main method.
- Console:** Displays the terminal output of the test execution, showing log messages from Selenium and the test results ("Login Success" and "Side Menu Working").
- Bottom Bar:** Includes a search bar, application icons, and system status indicators (weather, battery, time).

```
19
20     WebElement password=driver.findElement(By.id("password"));
21     WebElement login=driver.findElement(By.xpath("//[@id='login-button']"));
22     username.sendKeys("standard_user");
23     password.sendKeys("secret_sauce");
24     login.click(); String actualUrl="https://www.saucedemo.com/";
25     String expectedUrl= driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl))
26         System.out.println("Incorrect Login Details");
27     } else {
28         System.out.println("Login Success");
29     }
30
31     ((JavascriptExecutor) driver).executeScript("scroll(0,300)");
32     Actions ac = new Actions(driver);
33     WebElement wb = driver.findElement(By.xpath("//[@id='react-burger-menu-btn']"));
34     ac.moveToElement(wb).build().perform();
35     Thread.sleep(3000);
36     WebElement plan = driver.findElement(By.xpath("//[@id='react-burger-menu-btn']"));
37     if (wb.isDisplayed()) {
38         System.out.println("Side Menu Working");
39     } else {
40         System.out.println("Side Menu is not Working");
41     }
42     plan.click();
43     Thread.sleep(2000);
44     driver.findElement(By.xpath("//[@id='react-burger-menu-btn']")).click();
45     Thread.sleep(4000);
```

```
<terminated> testNavBar [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.core\openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe [Jul 8
Jul 08, 2022 8:34:50 PM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 08, 2022 8:34:53 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Login Success
Side Menu Working
```

## Test Case: -TC/10

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Normal
<b>Test Case ID</b>	TC/10
<b>Test Case</b>	Check whether users able to close to side menu with close button
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password</li><li>3. Click side menu bar.</li><li>4. Click close menu button</li></ol>
<b>Input data</b>	standard_user / secret_sauce
<b>Expected Results</b>	Side menu button should close using close button.
<b>Actual Results</b>	Side menu button closed using close button.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbar:** Standard toolbar icons for file operations like Open, Save, Print, etc.
- Package Explorer:** Shows two projects: "test" and "testCaseSauceDemo".
- Editor Area:** Displays the Java code for "nav\_crossBtn.java". The code uses Selenium WebDriver to interact with a browser. It includes methods like "display2()", "display3()", and "display4()". The "display2()" method scrolls the page, finds the burger menu, and checks if it's displayed. If it is, it prints "Side Menu Working"; if not, it prints "Side Menu is not Working". It then clicks the menu and sleeps for 2000ms before sleeping for 4000ms. The "display3()" method sets the chromedriver path and opens the saucedemo website. The "display4()" method is partially visible.
- Outline View:** Shows the class structure with methods: main(), display1(), display2(), display3(), and display4().
- Console Tab:** Shows the terminal output of the application. It includes log messages like "INFO: Detected upstream dialect: W3C", "Jul 09, 2022 8:27:37 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch", "INFO: Found exact CDP implementation for version 103", "Login Success", and "Side Menu Working".
- System Tray:** Shows system status icons including battery level (83%), network, volume, and date/time (7/9/2022).

## Test Case: -TC/11

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/11
<b>Test Case</b>	Check whether inventory page item text displayed.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the image title of inventory page.</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Image title should display on inventory page
<b>Actual Results</b>	Image title displayed on inventory page
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows a Java project structure with packages: test, seleniumTest, and Chrome Driver. Under test, there are several Java files: correct\_Logins.java, dddd.java, image\_text.java, imageCheck.java, incorrect\_password.java, incorrect\_UserPass.java, inventory\_img.java, invalid\_userName.java, loginPageUser.java, nav\_crossBtn.java, test\_navbar.java, and chromedriver.exe.
- Code Editor:** Displays the content of the image\_text.java file. The code uses Selenium WebDriver to navigate to the saucedemo website, enter credentials, and check if an element with id="item\_4\_title\_link" is displayed.
- Outline View:** Shows the class structure with the main method highlighted.
- Console:** Shows the terminal output of the Java application, indicating successful execution and log messages from the Selenium driver.
- Taskbar:** Shows the Windows taskbar with various pinned icons and system status.

## Test Case: -TC/12

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/12
<b>Test Case</b>	Check the inventory page image description displayed.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the item description of inventory page.</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Image title should display on inventory page
<b>Actual Results</b>	Image description displayed on inventory page
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Project Structure:** The Package Explorer view shows a project named "test" containing a "seleniumTest" package with several Java files: correct\_Logins.java, dddd.java, image\_text.java, imageCheck.java, img\_desc.java, incorrect\_password.java, incorrect\_UserPass.java, inventory\_img.java, invalid\_userName.java, loginPageUser.java, nav\_crossBtn.java, test\_navBar.java, and chromedriver.exe.
- Code Editor:** The main editor window displays the content of the "img\_desc.java" file. The code is a Java class named "img\_desc" with a main method that uses Selenium WebDriver to log in to the saucedemo website and verify the inventory page's image description.
- Output View:** The Console tab shows the terminal output of the application running. It includes logs from the Selenium WebDriver and the Java application, indicating successful login and the verification of the image description.
- System Tray:** The taskbar at the bottom right shows system icons and information like the date (7/9/2022), time (10:24 AM), and battery level (82%).

## Test Case: -TC/13

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of login page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/13
<b>Test Case</b>	Check the inventory page item price displayed.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the item price of inventory page</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Item price should display on inventory page
<b>Actual Results</b>	Item price displayed on inventory page
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar icons.
- Package Explorer:** Shows the project structure with packages like test, seleniumTest, and Chrome Driver, and files such as correct\_Logins.java, dddd.java, image\_text.java, imageCheck.java, img\_desc.java, incorrect\_password.java, incorrect\_UserPass.java, inventory\_img.java, item\_price.java, invalid\_userName.java, loginPageUser.java, nav\_crossBtn.java, testNavBar.java, and chromedriver.exe.
- Editor:** Displays the Java code for the item\_price class. The code imports various Selenium and Java libraries, sets up a WebDriver, and performs a login operation to check if the actual URL matches the expected URL.
- Outline View:** Shows the class structure with main(String[] args) method.
- Console:** Shows the terminal output of the Java application, indicating successful execution and a true value for the item price.
- Taskbar:** Shows various open applications including a browser icon.
- System Tray:** Shows system status icons like battery level, network, and date/time.

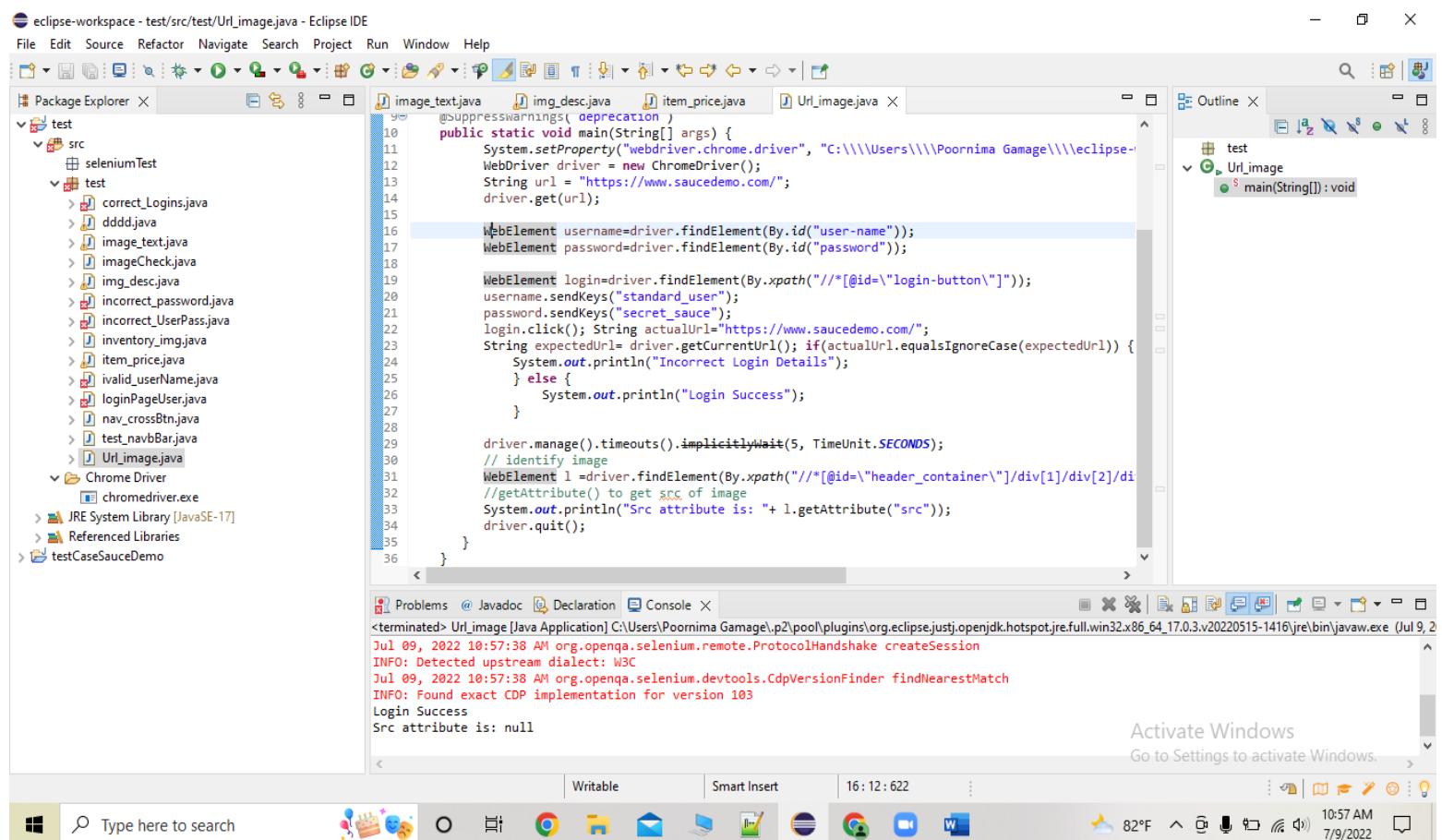
## Test Case: -TC/14

**Module:** <https://www.saucedemo.com>

## Sprint: Sprint 4

**Author:** L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/14
<b>Test Case</b>	Check the inventory page logo directed to saucedemo home page.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the logo url</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Inventory page logo should redirected to saucedemo web site home page.
<b>Actual Results</b>	Inventory page logo is not redirected to saucedemo web site home page.
<b>Status</b>	fail
<b>Comment</b>	Image 01 is shared for reference



## Test Case: -TC/15

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/15
<b>Test Case</b>	Check whether the inventory product image displayed correctly.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>4. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>5. Login the saucedemo web site using correct username and password.</li><li>1. Check the product image</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Inventory product image should displayed correctly
<b>Actual Results</b>	Inventory product image displayed correctly
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar icons.
- Left Sidebar:** Package Explorer (test), Problems, Javadoc, Declaration, Console.
- Central Area:** Code editor showing `inventory_img.java` file content:

```
1 package test;
2
3 import org.openqa.selenium.By;
4
5 public class inventory_img {
6
7
8     public static void main(String[] args) {
9         System.setProperty("webdriver.chrome.driver", "C:\\Users\\Poornima Gamage\\eclipse-workspace\\test\\chromedriver.exe");
10        WebDriver driver=new ChromeDriver();
11        driver.manage().window().maximize();
12        driver.get("https://www.saucedemo.com/");
13
14        WebElement username=driver.findElement(By.id("user-name"));
15        WebElement password=driver.findElement(By.id("password"));
16
17        WebElement login=driver.findElement(By.xpath("//*[@id='login-button']"));
18        username.sendKeys("standard_user");
19        password.sendKeys("secret_sauce");
20        login.click(); String actualUrl="https://www.saucedemo.com/";
21        String expectedUrl= driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl)) {
22            System.out.println("Incorrect Login Details");
23        } else {
24            System.out.println("Login Success");
25        }
26        WebElement logo_img = driver.findElement(By.xpath("//*[@id='item_4_img_link']/img"));
27    }
28
29
30 }
```
- Right Sidebar:** Outline (test, inventory\_img, main(String[])).
- Bottom Status Bar:** Activate Windows, Go to Settings to activate Windows.
- Taskbar:** Shows the Windows Start button, a search bar, and pinned application icons for Edge, Google Chrome, Mail, File Explorer, Task View, and others.
- System Tray:** Shows battery level (86%), network status, and system icons.

# Test Case: -TC/16

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/16
<b>Test Case</b>	Check whether inventory filter drop down working properly.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the product filter list</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Inventory filter drop down should working properly.
<b>Actual Results</b>	Inventory filter drop down worked properly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Package Explorer:** Shows a project structure with a `test` package containing several Java files: `correct_Logins.java`, `dddd.java`, `image_text.java`, `imageCheck.java`, `img_desc.java`, `incorrect_password.java`, `incorrect_UserPass.java`, `inventory_filter.java`, `inventory_img.java`, `item_price.java`, `invalid_userName.java`, `loginPageUser.java`, `nav_crossBtn.java`, `testNavBar.java`, and `Url_image.java`.
- Editor:** Displays the `inventory_filter.java` file content:public class inventory\_filter {
 public static void main(String[] args) throws InterruptedException {
 System.setProperty("webdriver.chrome.driver", "C:\\\\Users\\\\Poornima Gamage\\\\eclipse
 // ChromeDriver c = new ChromeDriver();
 WebDriver driver = new ChromeDriver();
 driver.manage().window().maximize();
 driver.get("https://www.saucedemo.com/");

 WebElement username=driver.findElement(By.id("user-name"));
 WebElement password=driver.findElement(By.id("password"));

 WebElement login=driver.findElement(By.xpath("//\*[@id='login-button']"));
 username.sendKeys("standard\_user");
 password.sendKeys("secret\_sauce");
 login.click(); String actualUrl="https://www.saucedemo.com/";
 String expectedUrl=driver.getCurrentUrl(); if(actualUrl.equalsIgnoreCase(expectedUrl))
 System.out.println("Incorrect Login Details");
 } else {
 System.out.println("Login Success");
 }

 Select selectelement=new Select(driver.findElement(By.xpath("//\*[@id='header\_contain
 List
- Outline View:** Shows the `inventory_filter` class and its `main(String[])` method.
- Console:** Shows the output of the Java application:

```
INFO: Found exact CDP implementation for version 103
Login Success
4
The values are Name (A to Z)
The values are Name (Z to A)
The values are Price (low to high)
The values are Price (high to low)
```
- System Tray:** Shows icons for battery level (85%), signal strength, and system status.

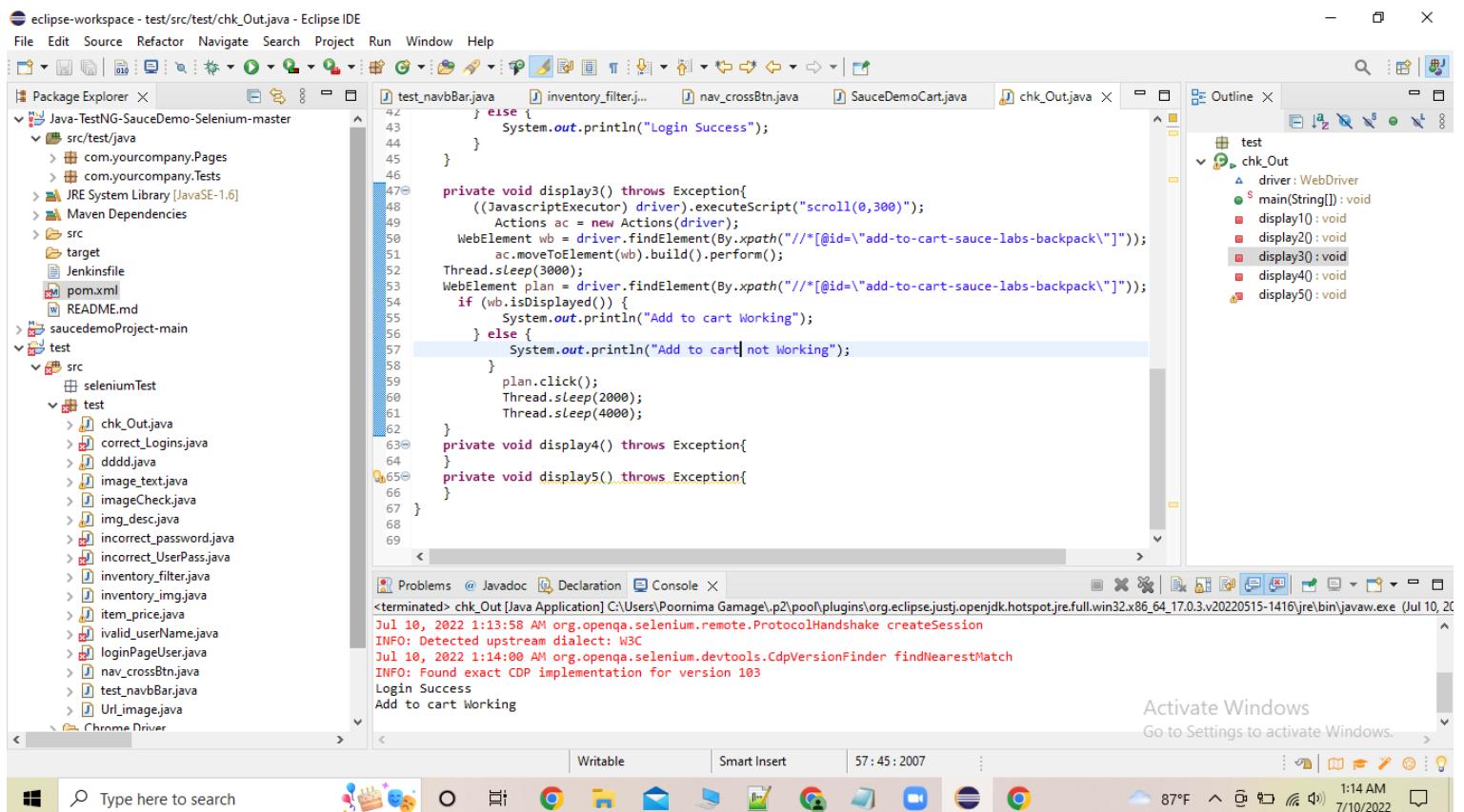
## Test Case: -TC/17

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/17
<b>Test Case</b>	Check whether Add to cart button working properly.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Check the add to cart button.</li> </ol>
<b>Input data</b>	My_username
<b>Expected Results</b>	Add to cart button should work properly.
<b>Actual Results</b>	Add to cart button worked properly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference



The screenshot shows the Eclipse IDE interface with the following details:

- Project Structure:** The "Package Explorer" view shows a project named "Java-TestNG-SauceDemo-Selenium-master". It contains several Java files under "src/test/java" and "src" (including "seleniumTest").
- Code Editor:** The main editor window displays the file "chk\_Out.java". The code is a Selenium test case using TestNG annotations. It includes methods like "display3()", "display4()", and "display5()" which interact with the browser using Actions and JavaScript.
- Outline View:** The "Outline" view on the right shows the class structure of "chk\_Out", including its methods and their implementations.
- Console:** The bottom "Console" tab shows the terminal output of the test execution. It includes log messages from Selenium and the browser, such as "Login Success" and "Add to cart Working".
- System Bar:** The bottom bar shows the Windows taskbar with various pinned icons and system status information.

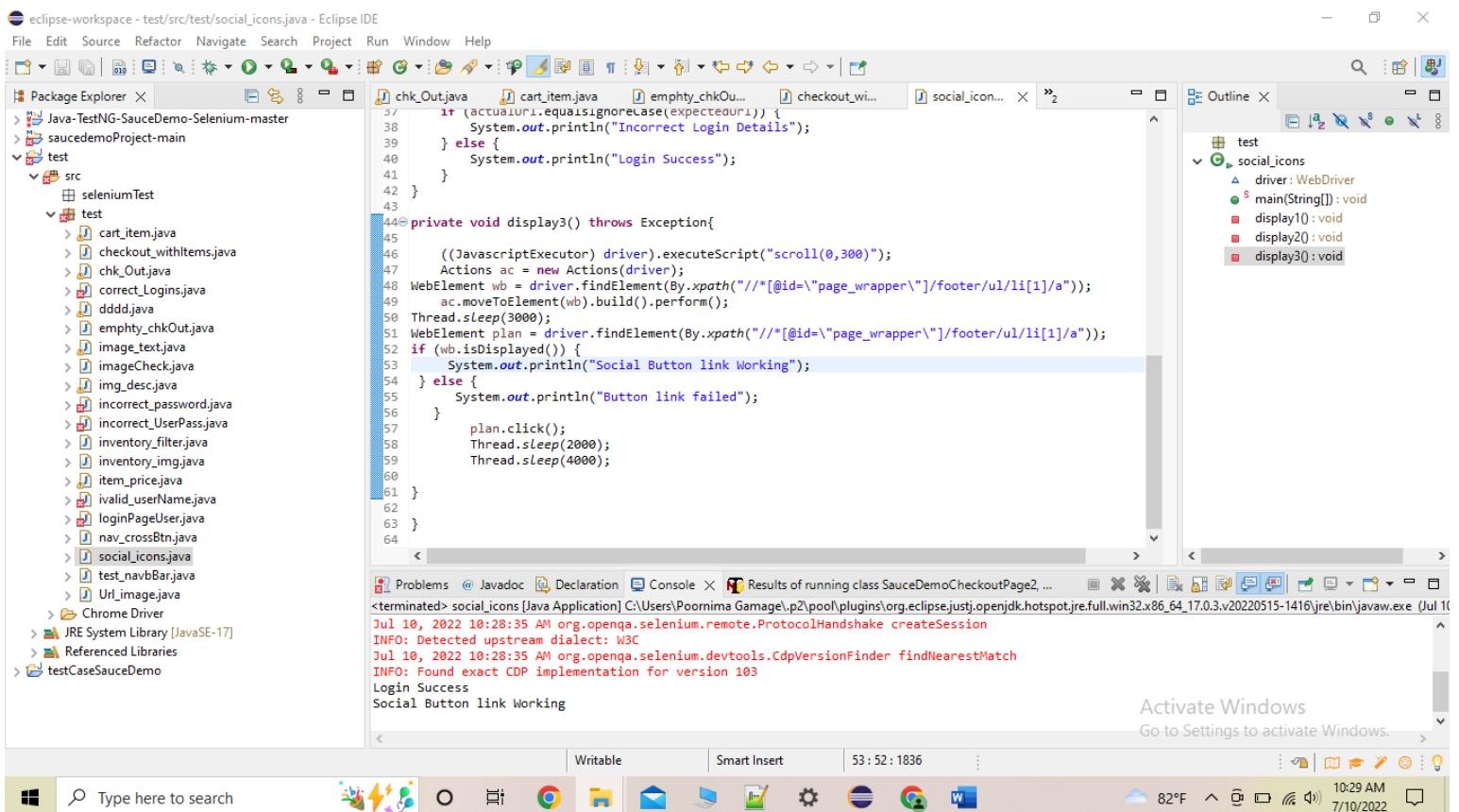
## Test Case: -TC/18

Module: <https://www.saucedemo.com>

Sprint: Sprint 4

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of inventory page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/18
<b>Test Case</b>	Check whether social button in footer working properly.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Check the social button.</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	social button in footer should work properly
<b>Actual Results</b>	social button in footer worked properly
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference



```

eclipse-workspace - test/src/test/social_icons.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X Outline X
Java-TestNG-SauceDemo-Selenium-master
saucedemoProject-main
test
src
  seleniumTest
    test
      cart_item.java
      checkout_withItems.java
      chk_Out.java
      correct_Logins.java
      dddd.java
      empty_chkOut.java
      image_text.java
      imageCheck.java
      img_desc.java
      incorrect_password.java
      incorrect_UserPass.java
      inventory_filter.java
      inventory_img.java
      item_price.java
      invalid_userName.java
      loginPageUser.java
      nav_crossBtn.java
      social_icons.java
      testNavBar.java
      Url_image.java
      Chrome Driver
      JRE System Library [JavaSE-17]
      Referenced Libraries
      testCaseSauceDemo
chk_Out.java
  int (actualuri.equalsignoreCase(expecturi)) {
  System.out.println("Incorrect Login Details");
} else {
  System.out.println("Login Success");
}
private void display3() throws Exception{
  ((JavascriptExecutor) driver).executeScript("scroll(0,300)");
  Actions ac = new Actions(driver);
  WebElement wb = driver.findElement(By.xpath("//*[@id='page_wrapper']/footer.ul/li[1]/a"));
  ac.moveToElement(wb).build().perform();
  Thread.sleep(3000);
  WebElement plan = driver.findElement(By.xpath("//*[@id='page_wrapper']/footer.ul/li[1]/a"));
  if (wb.isDisplayed()) {
    System.out.println("Social Button link Working");
  } else {
    System.out.println("Button link failed");
  }
  plan.click();
  Thread.sleep(2000);
  Thread.sleep(4000);
}

```

Problems @ Javadoc Declaration Console X Results of running class SauceDemoCheckoutPage2...
<terminated> social\_icons [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86\_64\_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 10, 2022 10:28:35 AM org.openqa.selenium.remote.ProtocolHandshake createSession
INFO: Detected upstream dialect: W3C
Jul 10, 2022 10:28:35 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Login Success
Social Button link Working

Activate Windows  
Go to Settings to activate Windows.

## Test Case: -TC/19

Module: <https://www.saucedemo.com/cart.html>

Sprint: Sprint 5

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/19
<b>Test Case</b>	Check whether cart button opens cart url properly.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the cart button.</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Cart button should opens cart url properly
<b>Actual Results</b>	Cart button opened the cart url properly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Structure:** The Package Explorer view shows a project named "Java-TestNG-SauceDemo-Selenium-master" containing packages like "saucedemoProject-main" and "test".
- Code Editor:** The main editor window displays Java code for a Selenium test case. The code includes methods for navigating to the homepage, logging in, and checking if the shopping cart is open. It uses WebDriver and Actions from the Selenium library.
- Outline View:** The Outline view on the right shows the class structure with methods: driver : WebDriver, main(String[]), display1(), display2(), display3(), display4(), and display5().
- Console:** The bottom console window shows the terminal output of the test execution, including log messages from Selenium and the browser's developer tools.
- Taskbar:** The taskbar at the bottom includes icons for search, file operations, and system status.

## Test Case: -TC/20

Module: <https://www.saucedemo.com/cart.html>

Sprint: Sprint 5

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/20
<b>Test Case</b>	Check whether checkout button work properly.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Check the checkout button.</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Check button should opens with items.
<b>Actual Results</b>	Checkout button can go with empty products.
<b>Status</b>	Fail
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar with icons for file operations, search, and project navigation.
- Package Explorer:** Shows the project structure with files like `nav_crossBtn.java`, `SauceDemoCart.java`, `chk_Out.java`, `cart_item.java`, and `empty_chkOut.java`.
- Code Editor:** Displays the Java code for `empty_chkOut.java`. The code implements a `WebDriver` interface to handle browser interactions, including logging in, navigating to the cart, and checking the checkout button.
- Outline View:** Shows the class structure with methods `display1()`, `display2()`, `display3()`, and `display4()`.
- Problems View:** Shows build errors and warnings.
- Console View:** Displays log output from the application run, including startup messages and test results.
- Taskbar:** Shows the current workspace name, file status (Writable), and other open tabs.
- System Tray:** Shows system status icons for battery, network, and system time (7:30 AM, 7/10/2022).

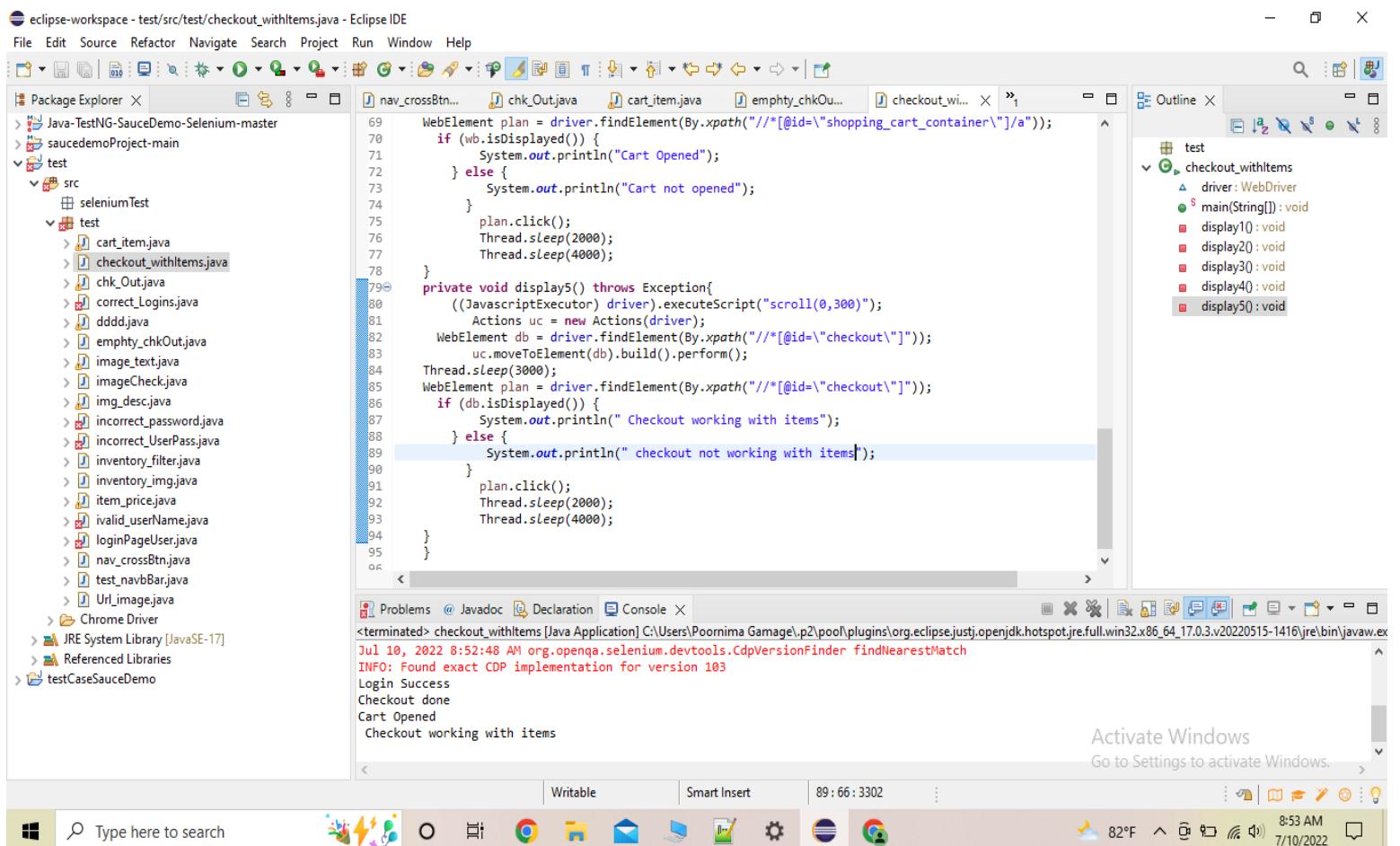
## Test Case: -TC/21

Module: <https://www.saucedemo.com/cart.html>

Sprint: Sprint 5

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/21
<b>Test Case</b>	Check whether checkout button work properly with product items
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Check the cart button.</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Cart button should display cart products
<b>Actual Results</b>	Cart button displayed the cart products.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference



The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure with files like `Java-TestNG-SauceDemo-Selenium-master`, `saucedemoProject-main`, and a `test` folder containing `seleniumTest` and `checkout_withItems.java`.
- Code Editor:** Displays the Java code for `checkout_withItems.java`. The code uses Selenium WebDriver to interact with a web application, specifically checking if the checkout button works correctly with items in the cart.
- Outline View:** Shows the class structure with methods `display1()`, `display2()`, `display3()`, `display4()`, and `display5()`.
- Console:** Shows the terminal output with messages like "Login Success", "Checkout done", "Cart Opened", and "Checkout working with items". It also includes a note about finding the exact CDP implementation for version 103.
- System Tray:** Shows system status icons including battery level (82%), temperature (853 AM), and date/time (7/10/2022).

```

69    WebElement plan = driver.findElement(By.xpath("//*[@id='shopping_cart_container']/a"));
70    if (wb.isDisplayed()) {
71        System.out.println("Cart Opened");
72    } else {
73        System.out.println("Cart not opened");
74    }
75    plan.click();
76    Thread.sleep(2000);
77    Thread.sleep(4000);
78}
private void display5() throws Exception{
79    ((JavascriptExecutor) driver).executeScript("scroll(0,300)");
80    Actions uc = new Actions(driver);
81    WebElement db = driver.findElement(By.xpath("//*[@id='checkout']"));
82    uc.moveToElement(db).build().perform();
83    Thread.sleep(3000);
84    WebElement plan = driver.findElement(By.xpath("//*[@id='checkout']"));
85    if (db.isDisplayed()) {
86        System.out.println(" Checkout working with items");
87    } else {
88        System.out.println(" checkout not working with items");
89    }
90    plan.click();
91    Thread.sleep(2000);
92    Thread.sleep(4000);
93}
94}
95}

```

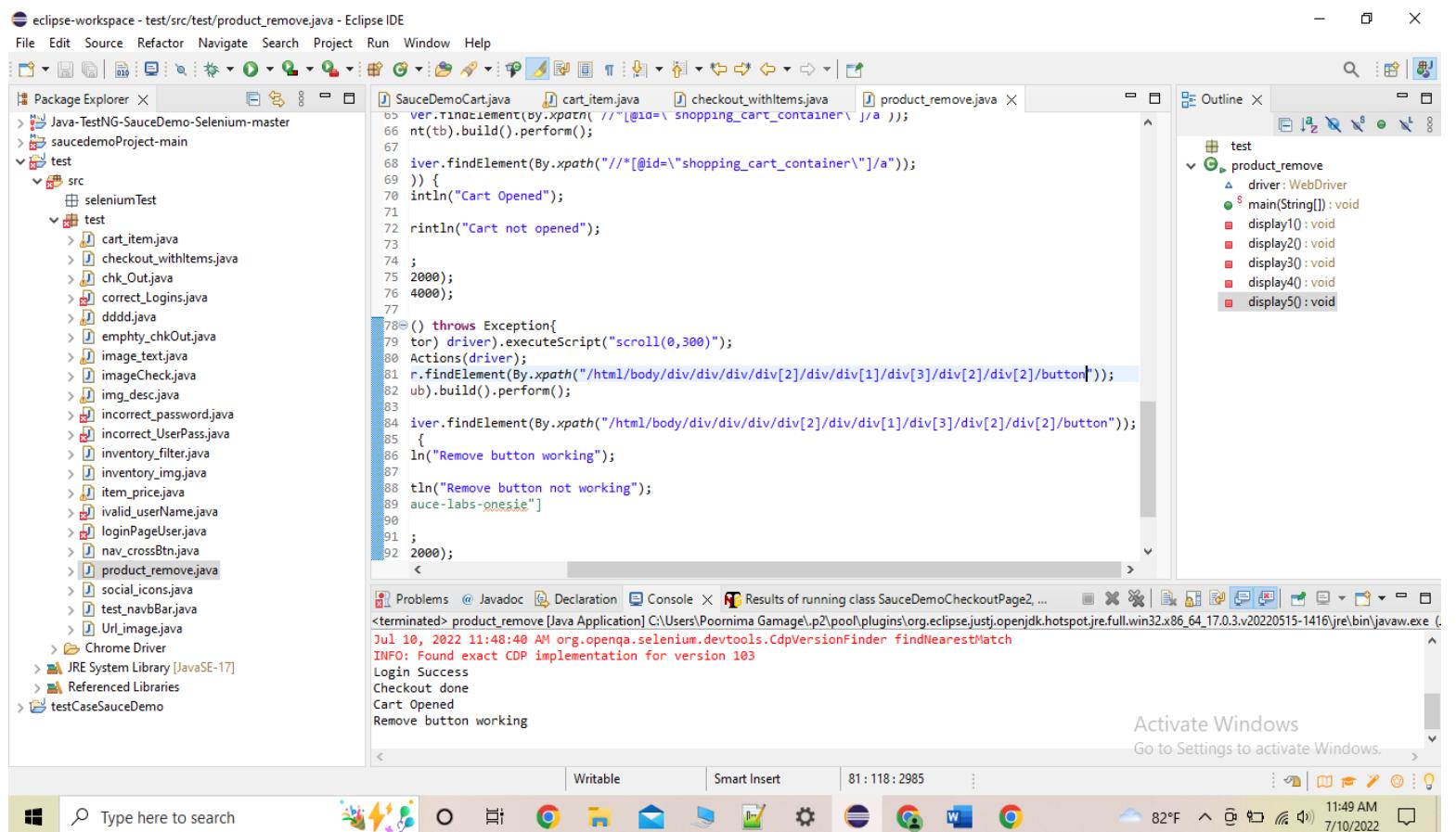
## Test Case: -TC/22

**Module:** <https://www.saucedemo.com/cart.html>

## Sprint: Sprint 5

**Author:** L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/21
<b>Test Case</b>	Check whether remove button work properly with product items.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Check the remove item button in the cart.</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	remove button should work properly with product items in cart page.
<b>Actual Results</b>	remove button worked properly with product items.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference



## Test Case: -TC/23

Module: <https://www.saucedemo.com/cart.html>

Sprint: Sprint 5

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/21
<b>Test Case</b>	Check whether continue shopping button work properly with product items.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Check the remove item button in the cart.</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	continue shopping button should work properly with product items in cart page.
<b>Actual Results</b>	continue shopping button worked properly with product items.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

```

eclipse-workspace - test/src/test/continue_shopping.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X SauceDemoCart.java cart_item.java checkout_with... product_remove... continue_shop...
74 System.out.println("Cart not opened");
75
76 plan1.click();
77 Thread.sleep(2000);
78 Thread.sleep(4000);
79
80 void display5() throws Exception{
81 JavascriptExecutor driver = ((JavascriptExecutor) driver);
82 Actions dc = new Actions(driver);
83 WebElement ub = driver.findElement(By.xpath("//*[@id='continue-shopping']"));
84 dc.moveToElement(ub).build().perform();
85 sleep(3000);
86 WebElement plan2 = driver.findElement(By.xpath("//*[@id='continue-shopping']"));
87 if(plan2.isDisplayed()){
88 System.out.println("Continue Shopping button working");
89 }
90 System.out.println("Continue Shopping not working");
91 WebElement remove = driver.findElement(By.id("remove-sauce-labs-one-size"));
92 remove.click();
93 Thread.sleep(2000);
94 Thread.sleep(4000);
95
96
97
98
99
100
101
    
```

Outline View:

- test
  - continue\_shopping
    - driver: WebDriver
    - main(String[]): void
    - display10(): void
    - display20(): void
    - display30(): void
    - display40(): void
    - display50(): void

Problems View:

```

<terminated> continue_shopping [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.core\1.17.0.v20220515-1416\jre\bin\javaw.exe
Jul 10, 2022 11:54:08 AM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch
INFO: Found exact CDP implementation for version 103
Login Success
Checkout done
Cart Opened
Continue Shopping button working
    
```

## Test Case: -TC/24

Module: <https://www.saucedemo.com/cart.html>

Sprint: Sprint 5

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of cart page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/21
<b>Test Case</b>	Check whether item price display on cart page.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Check the item price.</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	item price should display on cart page.
<b>Actual Results</b>	item price displayed on cart page.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

```

eclipse-workspace - test/src/test/getTotal.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X continue_shopping.java getTotal.java X image_text.java
src
  seleniumTest
    test
      cart_item.java
      checkout_withItems.java
      chk_Out.java
      continue_shopping.java
      correct_Logins.java
      correct_Password.java
      dddd.java
      empty_chkOut.java
      getTotal.java
      image_text.java
      imageCheck.java
      img_desc.java
      incorrect_password.java
      incorrect_UserPass.java
      inventory_filter.java
      inventory_img.java
      item_price.java
      invalid_userName.java
      loginPageUser.java
      nav_crossBtn.java
      product_remove.java
      social_icons.java
      test_navbar.java
      Url_image.java
  Chrome Driver
  JRE System Library [JavaSE-17]
  Referenced Libraries
testCaseSauceDemo

private void display4() throws Exception{
    ((JavascriptExecutor) driver).executeScript("scroll(0,300)");
    Actions ec = new Actions(driver);
    WebElement tb = driver.findElement(By.xpath("//*[@id='shopping_cart_container']/a"));
    ec.moveToElement(tb).build().perform();
    Thread.sleep(3000);
    WebElement plan1 = driver.findElement(By.xpath("//*[@id='shopping_cart_container']/a"));
    if (tb.isDisplayed()) {
        System.out.println("Cart Opened");
    } else {
        System.out.println("Cart not opened");
    }
    plan1.click();
    Thread.sleep(2000);
    Thread.sleep(4000);
}
private void display5() throws Exception{
    String Display = driver.findElement(By.xpath("//*[@id='cart_contents_container']/div/div")).getText();
    //To print the value
    System.out.println("Total displayed is :" +Display);
}

main(String[] args) {
    display4();
    display5();
}

```

Output window:

```

<terminated> getTotal [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 10, 2022, 12:08 PM)
Login Success
Checkout done
Cart Opened
Total displayed is :$29.99

```

System tray:

```

Activate Windows
Go to Settings to activate Windows.
12:08 PM 7/10/2022
82°F

```

## Test Case: -TC/25

Module: <https://www.saucedemo.com/checkout-step-one.html>

Sprint: Sprint 6

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step one page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/25
<b>Test Case</b>	Check whether customer details submission works properly.
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Click checkout on cart page</li> <li>5. Check customer informations</li> </ol>
<b>Input data</b>	Nipuna
<b>Expected Results</b>	customer details should not submit without empty values
<b>Actual Results</b>	customer details unable submitted with empty values
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

```

eclipse-workspace - test/src/test/cus_details.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X getTotals.java image_text.java cus_details.java checkout_withItems.java
Java-TestNG-SauceDemo-Selenium-master
saucedemoProject-main
test
src
  seleniumTest
    test
      cart_item.java
      checkout_withItems.java
      chk_Out.java
      continue_shopping.java
      correct_Logins.java
      cus_details.java
      dddd.java
      emptyt_chkOut.java
      getTotal.java
      image_text.java
      imageCheck.java
      img_desc.java
      incorrect_password.java
      incorrect_UserPass.java
      inventory_filter.java
      inventory_img.java
      item_price.java
      invalid_userName.java
      loginPageUser.java
      nav_crossBtn.java
      product_remove.java
      social_icons.java
      test_navBar.java
      Url_image.java
      Chrome Driver
      JRE System Library [JavaSE-17]
      Referenced Libraries
      TestCaserSeleniumName
getTotals.java
image_text.java
cus_details.java
checkout_withItems.java
Outline X
test
  cus_details
    driver : WebDriver
    $ main(String[])
    $ display10 : void
    $ display20 : void
    $ display30 : void
    $ display40 : void
    $ display50 : void
    $ display60 : void
Problems Declaration Console X Results of running class SauceDemoCheckoutPage2...
<terminated> cus_details [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 10, 2022, 11:44:39 PM)
Checkout done
Cart Opened
Checkout working with items
Details not submitted sucessfully
Activate Windows
Go to Settings to activate Windows.
9:16 PM 82°F 7/10/2022
Type here to search

```

## Test Case: -TC/26

Module: : <https://www.saucedemo.com/checkout-step-one.html>

Sprint: Sprint 6

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step one page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/26
<b>Test Case</b>	Check whether customer details submission works properly.
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Select product and click Add to cart</li><li>4. Click checkout on cart page</li><li>5. Check customer informations</li></ol>
<b>Input data</b>	Nipuna,Sanjaya,60600
<b>Expected Results</b>	customer details should submit with all values
<b>Actual Results</b>	customer details submitted with all values
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure with a package named "test" containing several Java files: getTotal.java, image\_text.java, cus\_details.java, and checkout\_withItems.java.
- Code Editor:** Displays the cus\_details.java file. The code is a Selenium test script using WebDriver. It includes methods like display6() which handles user input for first name, last name, and zip code, and then performs a click operation on a continue button. It also checks the actual URL against an expected URL.
- Outline View:** Shows the class structure with methods: main(), display1(), display2(), display3(), display4(), display5(), and display6().
- Console:** Shows the output of the test execution, indicating that the checkout process worked successfully.
- Taskbar:** Shows the current time as 9:14 PM and the date as 7/10/2022.

## Test Case: -TC/27

**Module:** : <https://www.saucedemo.com/checkout-step-two.html>

## Sprint: Sprint 7

**Author:** L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step two page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/27
<b>Test Case</b>	Check whether payment information displayed correctly
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Click checkout on cart page</li> <li>5. Check customer informations</li> <li>6. Check final checkout informations</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	payment information should display correctly
<b>Actual Results</b>	payment information displayed correctly
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** eclipse-workspace - test/src/test/totalMoney.java - Eclipse IDE
- Menu Bar:** File Edit Source Refactor Navigate Search Project Run Window Help
- Toolbars:** Standard, Java, Text Editors, Navigator, Outline, Problems, Javadoc, Declaration, Console.
- Left Sidebar (Package Explorer):** Shows Java TestNG SauceDemo-Selenium-master, saucedemoProject-main, and a test project with a src folder containing various test classes like cart\_item.java, checkout\_withItems.java, etc.
- Central Area (Code Editor):** Displays the totalMoney.java file with Java code for Selenium tests. The code includes methods for displaying user information and printing payment details.
- Right Sidebar (Outline View):** Shows the class structure for totalMoney, including methods like main(), display1() through display7().
- Bottom Status Bar:** Writable, Smart Insert, 120:112:4231.
- Bottom Taskbar:** Shows the results of running the class: Cart Opened, Checkout working with items, Details submitted sucessfully, and Payment information :SauceCard #31337.
- System Tray:** Shows the date and time as 10:02 PM 7/10/2022.

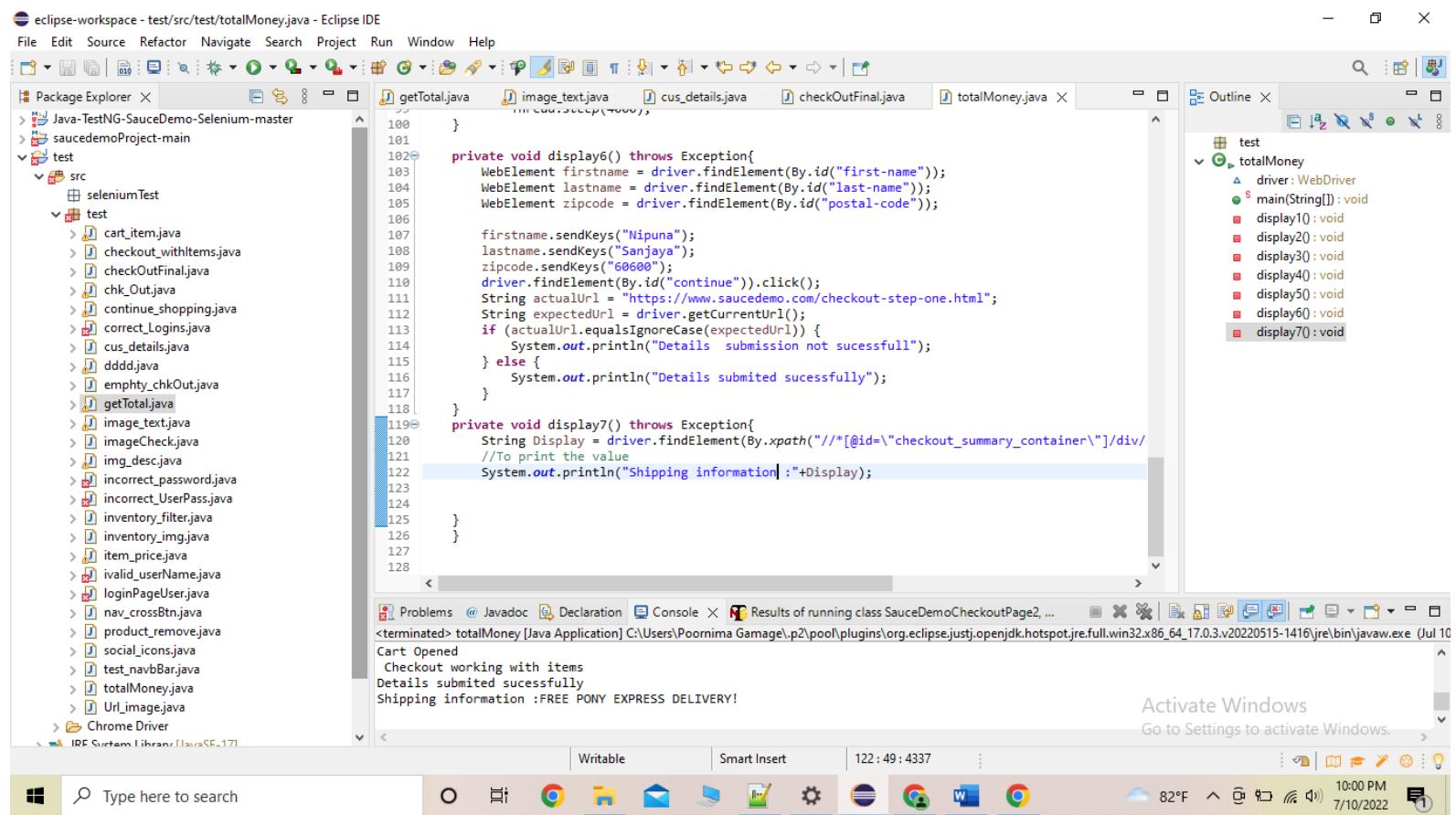
## Test Case: -TC/28

**Module:** : <https://www.saucedemo.com/checkout-step-two.html>

## Sprint: Sprint 7

**Author:** L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step two page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/28
<b>Test Case</b>	Check whether shipping information displayed correctly
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Click checkout on cart page</li> <li>5. Check customer informations</li> <li>6. Check final checkout informations</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	shipping information should display correctly
<b>Actual Results</b>	shipping information displayed correctly
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference



## Test Case: -TC/29

Module: : <https://www.saucedemo.com/checkout-step-two.html>

Sprint: Sprint 7

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step two page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/29
<b>Test Case</b>	Check whether Total amount without tax displayed correctly
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Click checkout on cart page</li> <li>5. Check customer informations</li> <li>6. Check final checkout informations</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Total amount without tax should display correctly
<b>Actual Results</b>	Total amount without tax displayed correctly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

```

eclipse-workspace - test/src/test/totalMoney.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X getTotals.java image_text.java cus_details.java checkOutFinal.java totalMoney.java X
src
  seleniumTest
    test
      cart_item.java
      checkout_withItems.java
      checkOutFinal.java
      chk_Out.java
      continue_shopping.java
      correct_Logins.java
      cus_details.java
      dddd.java
      empty_chkOut.java
      getTotal.java
      image_text.java
      imageCheck.java
      img_desc.java
      incorrect_password.java
      incorrect_UserPass.java
      inventory_filter.java
      inventory_img.java
      item_price.java
      invalid_userName.java
      loginPageUser.java
      nav_crossBtn.java
      product_remove.java
      social_icons.java
      test_navBar.java
      totalMoney.java
      Url_image.java
      Chrome Driver
Outline X
  test
    totalMoney
      driver : WebDriver
      main(String[])
      display10 : void
      display20 : void
      display30 : void
      display40 : void
      display50 : void
      display60 : void
      display70 : void
Problems @ Javadoc Declaration Console X Results of running class SauceDemoCheckoutPage2...
<terminated> totalMoney [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.core\1.17.0.v20220515-1416\jre\bin\javaw.exe (Jul 10)
Cart Opened
Checkout working with items
Details submited sucessfully
Total without tax displayed is :Item total: $29.99
Activate Windows
Go to Settings to activate Windows.
Type here to search 9:58 PM 82°F 7/10/2022

```

The screenshot shows the Eclipse IDE interface with the following details:

- Project Structure:** The Package Explorer shows a project named "Java-TestNG-SauceDemo-Selenium-master" containing a "saucedemoProject-main" module and a "test" package. The "test" package contains several Java files: cart\_item.java, checkout\_withItems.java, checkOutFinal.java, chk\_Out.java, continue\_shopping.java, correct\_Logins.java, cus\_details.java, dddd.java, empty\_chkOut.java, getTotal.java, image\_text.java, imageCheck.java, img\_desc.java, incorrect\_password.java, incorrect\_UserPass.java, inventory\_filter.java, inventory\_img.java, item\_price.java, invalid\_userName.java, loginPageUser.java, nav\_crossBtn.java, product\_remove.java, social\_icons.java, test\_navBar.java, totalMoney.java, Url\_image.java, and Chrome Driver.
- Code Editor:** The central editor window displays the content of the "getTotal.java" file. The code uses WebDriver to interact with a browser, specifically targeting the "checkout\_summary\_container" element to extract the total amount without tax.
- Output View:** The bottom right shows the "Console" output, which indicates that the test has run successfully, opening a cart, performing a checkout, and verifying that the total amount without tax is displayed correctly as "Item total: \$29.99".
- System Tray:** The taskbar at the bottom right shows system icons for battery level (82°F), network, and date/time (7/10/2022).

## Test Case: -TC/30

Module: : <https://www.saucedemo.com/checkout-step-two.html>

Sprint: Sprint 7

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step two page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/30
<b>Test Case</b>	Check whether tax displayed correctly
<b>Test Steps</b>	<ol style="list-style-type: none"> <li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li> <li>2. Login the saucedemo web site using correct username and password.</li> <li>3. Select product and click Add to cart</li> <li>4. Click checkout on cart page</li> <li>5. Check customer informations</li> <li>6. Check final checkout informations</li> </ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Total amount of tax should display correctly
<b>Actual Results</b>	Total amount of tax displayed correctly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

```

eclipse-workspace - test/src/test/totalMoney.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer X
Java-TestNG-SauceDemo-Selenium-master
saucedemoProject-main
test
src
  seleniumTest
    test
      cart_item.java
      checkout_withItems.java
      checkOutFinal.java
      chk_Out.java
      continue_shopping.java
      correct_Logins.java
      cus_details.java
      dddd.java
      empty_chkOut.java
      getTotal.java
      image_text.java
      imageCheck.java
      img_desc.java
      incorrect_password.java
      incorrect_UserPass.java
      inventory_filter.java
      inventory_img.java
      item_price.java
      invalid_userName.java
      loginPageUser.java
      nav_crossBtn.java
      product_remove.java
      social_icons.java
      test_navBar.java
      totalMoney.java
      Url_image.java
      Chrome Driver
      IRE System Library (JavaSE-17)
getTotal.java
image_text.java
cus_details.java
checkOutFinal.java
totalMoney.java
100
101
102() throws Exception{
103 name = driver.findElement(By.id("first-name"));
104 name = driver.findElement(By.id("last-name"));
105 zip = driver.findElement(By.id("postal-code"));
106
107 "/s("Nipuna");
108 s("Sanjaya");
109 ("600600");
110 it(By.id("continue")).click();
111 "https://www.saucedemo.com/checkout-step-one.html";
112 l = driver.getCurrentUrl();
113 alsIgnoreCase(expectedUrl) {
114 intln("Details submission not sucessfull");
115
116 intln("Details submited sucessfully");
117
118
119() throws Exception{
120 driver.findElement(By.xpath("//*[@id=\"checkout_summary_container\"]/div/div[2]/div[6]")).getText()
121 lue
122 ln("Total displayed is :"Display);
123
124
125
126
127
128
Problems @ Javadoc Declaration Console <terminated> totalMoney [Java Application] C:\Users\Poomima Gamage\p2\pool\plugins\org.eclipse.jdt.core\1.17.0.v20220515-1416\jre\bin\javaw.exe (Jul 10
Cart Opened
Checkout working with items
Details submitted sucessfully
Total displayed is :Tax: $2.40
Activate Windows
Go to Settings to activate Windows.
Type here to search
82°F 9:34 PM 7/10/2022

```

## Test Case: -TC/31

Module: : <https://www.saucedemo.com/checkout-step-two.html>

Sprint: Sprint 7

Author: L.G.N.S. Gamage

<b>Scenario</b>	Check the functionality of checkout step two page.
<b>Priority</b>	Major
<b>Test Case ID</b>	TC/31
<b>Test Case</b>	Check whether Total amount with tax displayed correctly
<b>Test Steps</b>	<ol style="list-style-type: none"><li>1. Go to <a href="https://www.saucedemo.com">https://www.saucedemo.com</a> web URL</li><li>2. Login the saucedemo web site using correct username and password.</li><li>3. Select product and click Add to cart</li><li>4. Click checkout on cart page</li><li>5. Check customer informations</li><li>6. Check final checkout informations</li></ol>
<b>Input data</b>	N/A
<b>Expected Results</b>	Total amount with tax should display correctly
<b>Actual Results</b>	Total amount with tax displayed correctly.
<b>Status</b>	pass
<b>Comment</b>	Image 01 is shared for reference

The screenshot shows the Eclipse IDE interface with the following details:

- File Menu:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Toolbar:** Standard Eclipse toolbar icons.
- Package Explorer:** Shows the project structure:
  - Java-TestNG-SauceDemo-Selenium-master
  - saucedemoProject-main
  - test
  - src
    - seleniumTest
    - test
      - cart\_item.java
      - checkout\_withItems.java
      - checkOutFinal.java
      - chk\_Out.java
      - continue\_shopping.java
      - correct\_Logins.java
      - cus\_details.java
      - dddd.java
      - empty\_chkOut.java
      - getTotal.java
      - image\_text.java
      - imageCheck.java
      - img\_desc.java
      - incorrect\_password.java
      - incorrect\_UserPass.java
      - inventory\_filter.java
      - inventory\_img.java
      - item\_price.java
      - invalid\_userName.java
      - loginPageUser.java
      - nav\_crossBtn.java
      - product\_remove.java
      - social\_icons.java
      - test\_navBar.java
      - totalMoney.java
      - Url\_image.java
    - Chrome Driver
  - Code Editor:** Displays the content of `totalMoney.java`. The code is a Selenium test case using TestNG annotations. It includes methods for entering names, selecting a shipping method, and checking the total price.
  - Outline View:** Shows the class structure:
    - test
    - totalMoney
      - driver : WebDriver
      - main(String[]) : void
      - display10 : void
      - display20 : void
      - display30 : void
      - display40 : void
      - display50 : void
      - display60 : void
      - display70 : void
  - Console:** Shows the output of the test execution:

```
<terminated> totalMoney [Java Application] C:\Users\Poornima Gamage\p2\pool\plugins\org.eclipse.jdt.openjdk.hotspot.jre.full.win32.x86_64_17.0.3.v20220515-1416\jre\bin\javaw.exe (Jul 10)
Cart Opened
Checkout working with items
Details submitted sucessfully
Total displayed is :32.39
```
  - System Tray:** Shows the Windows taskbar with various pinned icons.



## UNIVERSITY OF VOCATIONAL TECHNOLOGY

### SRI LANKA

#### Assignment Template & Feedback Form

Course Title	Software Testing and Reliability	Module	Software Testing and Reliability (IT406061)
Student's Name	L.G.N.S.Gamage	Batch No.	B2
Reg. No.	BED/19/B2/08	Semester	04
Resource Person	Ms. P.A. L. Madhushika.	Assignment No.	03
Issued on	20 / 06 / 2022		
Submission on	11 / 07 / 2022	Received by	

#### Feedback from the resource Person:

Key Strength :
To Improve Report:
General Comments:

Grade:

Date: .....

Resource Person's Signature: .....

Course Coordinator 's Signature: .....

