

CC1:

1)

```
package sample1.testsample;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;
import org.testng.Assert;
import org.testng.annotations.Test;
import io.github.bonigarcia.wdm.WebDriverManager;

public class cc1_1 {

    WebDriver driver;

    @Test
    public void f() throws InterruptedException {

        WebDriverManager.chromedriver().setup();
        ChromeOptions co=new ChromeOptions();
        co.addArguments("--remote-allow-origins=*");
        WebDriver driver=new ChromeDriver(co);
        driver.get("https://www.saucedemo.com/");
        driver.manage().window().maximize();

        Thread.sleep(5000);
        WebElement click1 = driver.findElement(By.id("user-name"));
        click1.click();
        click1.sendKeys("standard_user");
        WebElement click2 = driver.findElement(By.id("password"));
        click2.click();
        click2.sendKeys("secret_sauce");
        WebElement click3 = driver.findElement(By.id("login-button"));
        click3.click();
    }
}
```

```
Thread.sleep(5000);

WebElement click7 = driver.findElement(By.className("inventory_item_name"));

click7.getText();

System.out.println(click7.getText());

WebElement click4 = driver.findElement(By.id("add-to-cart-sauce-labs-backpack"));

click4.click();

WebElement click5 = driver.findElement(By.id("shopping_cart_container"));

click5.click();

WebElement click6 = driver.findElement(By.className("inventory_item_name"));

click6.getText();

System.out.println(click6.getText());

Assert.assertEquals(click6.getText(), click7.getText());

WebElement click8 = driver.findElement(By.className("cart_item"));

assert click8 != null;

WebElement c9 = driver.findElement(By.id("checkout"));

c9.click();

Thread.sleep(5000);

WebElement c1 = driver.findElement(By.id("first-name"));

c1.click();

c1.sendKeys("Ram");

WebElement c2 = driver.findElement(By.id("last-name"));

c2.click();

c2.sendKeys("Kumar");

WebElement c3 = driver.findElement(By.id("postal-code"));

c3.click();

c3.sendKeys("638416");

Thread.sleep(5000);

WebElement c4 = driver.findElement(By.id("continue"));

c4.click();
```

```
Thread.sleep(5000);

WebElement c5 = driver.findElement(By.className("inventory_item_name"));

System.out.println("Name of product :" + c5.getText());

WebElement c6 = driver.findElement(By.className("inventory_item_price"));

System.out.println("price of the product :" + c6.getText());

String c7 = driver.getTitle();

String c8 = driver.getCurrentUrl();

Assert.assertEquals(c7, "Swag Labs");

Assert.assertEquals(c8, "https://www.saucedemo.com/checkout-step-two.html");

}

}
```

```
<terminated> cc1.1 [TestNG] C:\Users\Irahu\Downloads\eclipse-jee-2022-09-R-win32-x86_64\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.4
[RemoteTestNG] detected TestNG version 7.4.0
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/56158(#936)) on port 39739
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
Name of product :Sauce Labs Backpack
price of the product :$29.99
PASSED: f

=====
      Default test
      Tests run: 1, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
=====
```

2)

```
package sample1.testsample;

import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.openqa.selenium.chrome.ChromeOptions;
import org.openqa.selenium.support.ui.Select;
//import org.testng.Assert;
import org.testng.annotations.Test;

import io.github.bonigarcia.wdm.WebDriverManager;

public class cc1_2 {

    @Test

    public void f() throws InterruptedException {

        WebDriverManager.chromedriver().setup();

        ChromeOptions co=new ChromeOptions();
        co.addArguments("--remote-allow-origins=*");

        WebDriver driver=new ChromeDriver(co);
        driver.get("https://www.saucedemo.com/");
        driver.manage().window().maximize();

        Thread.sleep(5000);

        WebElement click1 = driver.findElement(By.id("user-name"));
        click1.click();
        click1.sendKeys("standard_user");
```

```
WebElement click2 = driver.findElement(By.id("password"));

click2.click();

click2.sendKeys("secret_sauce");


WebElement click3 = driver.findElement(By.id("login-button"));

click3.click();


Thread.sleep(5000);


Select dropdown = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown.selectByVisibleText("Name (A to Z)");


Select dropdown1 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown1.selectByVisibleText("Name (Z to A)");


System.out.println("User is able to apply filter name wise from A-Z and Z-A");


Select dropdow = new
Select(driver.findElement(By.className("product_sort_container")));

dropdow.selectByVisibleText("Name (A to Z)");


WebElement c1 = driver.findElement(By.className("inventory_item_name"));

System.out.println("Name of the 1st product while filtering from A-Z : " +
c1.getText());


Thread.sleep(5000);


Select dropdown2 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown2.selectByVisibleText("Name (Z to A)");
```

```
WebElement c2 = driver.findElement(By.className("inventory_item_name"));

System.out.println("Name of the 1st product while filtering from Z-A : " +
c2.getText());

Thread.sleep(5000);

Select dropdown3 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown3.selectByVisibleText("Price (low to high)");

Select dropdown3 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown3.selectByVisibleText("Price (high to low)");

System.out.println(" user is able to apply filter from low price to high price and high
price to low price");

Select dropdown6 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown6.selectByVisibleText("Price (low to high)");

WebElement c5 = driver.findElement(By.className("inventory_item_price"));

System.out.println("Name of the 1st product price while filtering from low to high : "
+ c5.getText());

Thread.sleep(5000);

Select dropdown4 = new
Select(driver.findElement(By.className("product_sort_container")));

dropdown4.selectByVisibleText("Price (high to low)");


WebElement c6 = driver.findElement(By.className("inventory_item_price"));

System.out.println("Name of the 1st product Price while filtering from high to low: "
+ c6.getText());


}

}
```

Name : Rahul L
Reg No : 727721eucs113

```
<terminated> cc1_2 [TestNG] C:\Users\lrahu\Downloads\eclipse-jee-2022-09-R-win32-x86_64\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32
[RemoteTestNG] detected TestNG version 7.4.0
SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See http://www.slf4j.org/codes.html#StaticLoggerBinder for further details.
Starting ChromeDriver 112.0.5615.49 (bd2a7bcb881c11e8cfe3078709382934e3916914-refs/branch-heads/5615@{#936}) on port 36368
Only local connections are allowed.
Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.
ChromeDriver was started successfully.
User is able to apply filter name wise from A-Z and Z-A
Name of the 1st product while filtering from A-Z :Sauce Labs Backpack
Name of the 1st product while filtering from Z-A :Test.allTheThings() T-Shirt (Red)
user is able to apply filter from low price to high price and high price to low price
Name of the 1st product price while filtering from low to high :$7.99
Name of the 1st product Price while filtering from high to low:$49.99
PASSED: f

=====
      Default test
    Tests run: 1, Failures: 0, Skips: 0
=====

=====
Default suite
Total tests run: 1, Passes: 1, Failures: 0, Skips: 0
=====
```