Srabone Raxit

```java
package Testing;

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 java.util.List;


public class HomePageAutomation {


    public static void main(String[] args) throws InterruptedException {

        System.setProperty("webdriver.chrome.driver", "C:\\Users\\hp\\libs\\chromedriver-win64\\chromedriver.exe");

        ChromeOptions options = new ChromeOptions();

        options.addArguments("--remote-allow-origins=*");

        WebDriver driver = new ChromeDriver(options);


        try {

            driver.get("http://productedge.fwh.is/views/home_page.php");

            driver.manage().window().maximize();

            Thread.sleep(2000);


            System.out.println("Testing price filtering...");

            WebElement minPriceField = driver.findElement(By.id("min-price"));
```

```java
WebElement maxPriceField = driver.findElement(By.id("max-price"));

WebElement filterButton = driver.findElement(By.xpath("//button[text()='Filter']"));


// Input price range

minPriceField.sendKeys("10");

maxPriceField.sendKeys("50");

filterButton.click();

Thread.sleep(2000);


// Verify filtered products

List<WebElement> filteredProducts = driver.findElements(By.cssSelector(".product"));

for (WebElement product : filteredProducts) {

    String priceText = product.findElement(By.cssSelector(".price")).getText();

    double price = Double.parseDouble(priceText.replace("Price: $", ""));

    if (price < 10 || price > 50) {

        System.out.println("Error: Product outside price range displayed: $" + price);

    } else {

        System.out.println("Product within range: $" + price);

    }

}


// Test 2: Searching for a product

System.out.println("\nTesting product search...");

WebElement searchInput = driver.findElement(By.id("search-input"));

WebElement searchButton = driver.findElement(By.xpath("//button[text()='Search']"));


// Search for a product
```

```java
String searchQuery = "Coke";

searchInput.sendKeys(searchQuery);

searchButton.click();

Thread.sleep(2000);


// Verify search results

List<WebElement> searchResults = driver.findElements(By.cssSelector(".product"));

for (WebElement product : searchResults) {

    String productName = product.findElement(By.tagName("h2")).getText();

    if (!productName.toLowerCase().contains(searchQuery.toLowerCase())) {

        System.out.println("Error: Irrelevant product in search results: " + productName);

    } else {

        System.out.println("Relevant product found: " + productName);

    }

}


// === Test 3: Adding a product to the cart ===

System.out.println("\nTesting add to cart...");

List<WebElement> addToCartButtons = driver.findElements(By.xpath("//button[@name='add_to_cart']"));


// Click the first 'Add to Cart' button

if (addToCartButtons.size() > 0) {

    WebElement firstProductButton = addToCartButtons.get(0);

    WebElement firstProduct = firstProductButton.findElement(By.xpath("./ancestor::section[contains(@class, 'product')]"));

    String productName = firstProduct.findElement(By.tagName("h2")).getText();
```

```java
                firstProductButton.click();

                Thread.sleep(2000);


                System.out.println("Added product to cart: " + productName);
            } else {
                System.out.println("No products available to add to cart.");
            }


        } catch (Exception e) {
            e.printStackTrace();
        } finally {
            // Close the browser
            driver.quit();
        }
    }
}
```