



MIT ART DESIGN & TECHNOLOGY UNIVERSITY

MIT College of Management (MITCOM), Pune

**PROGRAMME: MASTER OF COMPUTER APPLICATION
(MCA DS)**

CERTIFICATE

This is to certify that, Mr. Kaustubh Dhone has submitted a Practical Report on _____ to MIT – ADT University, Pune for the partial fulfillment of Master in Computer Application (Data Science) submitted during the academic year 2024-25.

PRN No. ADT23MGTM0934 MCA Year. II MCA Sem. – Four

Subject Incharge

Dr. Alkawati Magadum
HOD MCA

Dr. Sangita Phunde
Principal

External Examiner

Sign of Examiners:

1. _____

Internal Examiner

Sign of Examiners:

2. _____

MIT ART DESIGN & TECHNOLOGY UNIVERSITY

MIT College of Management (MITCOM), Pune

Sub:-

Name: -

Div: - MCA-DS Div B

PRN: -

Sr. No.	Practical Title	Date of Completion	Faculty Sign
1	Set up Selenium WebDriver with Java: Install JDK, IDE, and configure Selenium.		
2	Google Search Functionality: Write a script to search for a term on Google using Selenium WebDriver methods.		
3	Handling Web Elements: Automate web methods and web operational methods.		
4	Handling Web Elements: Automate web locators using CSS selector methods.		
5	Handling Web Elements: Automate WebDriver wait methods with multiple browsers.		
6	Handling Web Elements: Automate checkbox, confirmation buttons, and prompt alert messages.		
7	Handling Web Elements: Automate dropdown selection and radio button interaction.		
8	Handling Frames: Automate frames, iframes, and nested frames.		
9	Alerts and Popups: Interact with simple, confirmation, and prompt alerts using Selenium.		
10	Form Submission: Automate form filling and submission using Selenium.		
11	Browser Compatibility Testing: Execute the same Selenium test across different browsers.		
12	Drag and Drop Web Elements: Automate drag and drop functionality using Selenium.		
13	Screenshots and Ad Blocking: Automate taking screenshots and block ads using Selenium tools.		
14	Web Table Interaction: Open a web table from a URL and automate operations on it.		
15	TestNG Framework: Implement automation testing using TestNG and create a structured test suite.		

1.Set up Selenium WebDriver with Java: Install JDK, IDE, and configure Selenium.

Code :

```
package Programs;
```

```
import org.openqa.selenium.WebDriver;
```

```
import
```

```
org.openqa.selenium.chrome.ChromeDriver;
```

```
public class Program {    public static void
```

```
main(String[] args) {
```

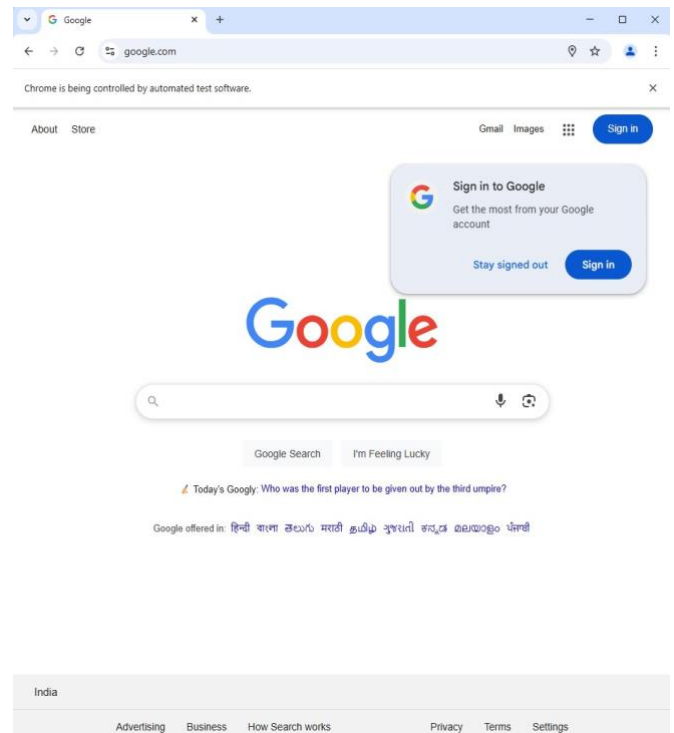
```
System.setProperty("webdriver.chrome.driver", "D:\\\\programs\\\\chromedriver-  
win64\\\\chromedriver.exe"); // Updated path
```

```
    WebDriver cd = new ChromeDriver();
```

```
cd.get("http://www.google.com");
```

```
    }
```

```
}
```



2 Write a Java Selenium script that to automate for Google Search Functionality: Write a script to search for a term on Google, handle all webdriver methods.

Code :

```
package Programs; import
org.openqa.selenium.By; import
org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
public class GoogleSearch{ public static
void main(String[] args) {

System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
    WebDriver cd = new
ChromeDriver();
cd.manage().window().maximize();
cd.get("https://www.google.com");
```

```
        WebElement searchBox =
cd.findElement(By.name("q"));
```

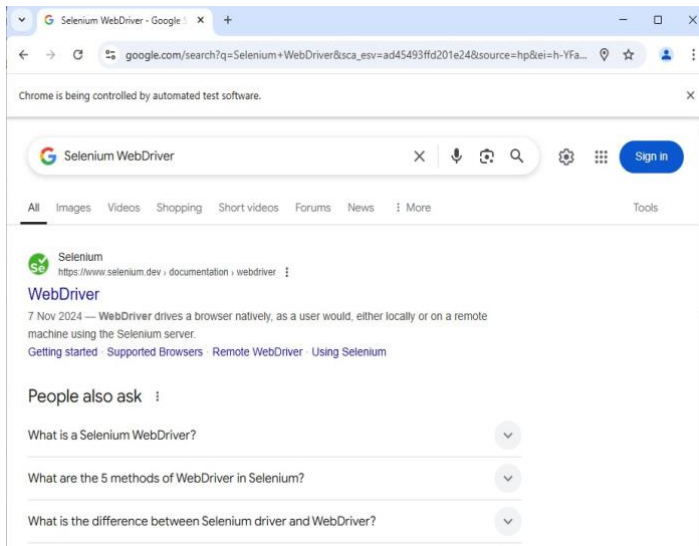
```
        searchBox.sendKeys("Selenium
WebDriver");    searchBox.submit();

        try {
            Thread.sleep(3000); // wait 3 seconds
        } catch (InterruptedException e) {
            e.printStackTrace();
        }

        System.out.println("Page Title: " +
cd.getTitle());

        // Don't close the browser
        immediately

        // cd.quit(); // <-- Commented out
    }
} //WhatUP MCA
```



```
// Create WebDriver instance
```

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

```
// Maximize browser window
```

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

```
// === Web Operational Methods ===
```

```
driver.get("https://demoqa.com/textbox"); //
Open demo site
```

3 Write a Java Selenium script that to automate for Handling Web Elements:to Automate web methods and web operational methods.

Code : package

Programs;

```
import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
```

```
public class Elements {    public static void
main(String[] args) {
```

```
    // Set path to ChromeDriver
```

```
System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

```
System.out.println("Title: " +
driver.getTitle());
```

```
System.out.println("URL: " +
driver.getCurrentUrl());
```

```
// === Web Element Methods ===
```

```
// 1. Find and fill the Full Name
input field
```

```
WebElement fullName =
driver.findElement(By.id("userName"));
```

```
fullName.sendKeys("Kamlesh
Bobade");
```

```
// 2. Email field
```

```
WebElement email =
driver.findElement(By.id("userEmail"));
```

```
email.sendKeys("kamlesh@example.com"
);
```

```

// 3. Current Address field

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

currentAddress.sendKeys("Pune,
Maharashtra");

// 4. Permanent Address field

WebElement permanentAddress =
driver.findElement(By.id("permanentAddr ess"));

permanentAddress.sendKeys("Aurangaba d,
Maharashtra");

// 5. Check visibility

if (fullName.isDisplayed() &&
fullName.isEnabled()) {

    System.out.println("Full Name field is
visible and enabled.");

}

// 6. Click the Submit button

WebElement submitButton =
driver.findElement(By.id("submit"));
submitButton.click();

// Pause for visual

try {

```

```

Thread.sleep(3000);    } catch
(InterruptedExceptio e) {

    e.printStackTrace();

}

// 7. Navigate example

driver.navigate().to("https://www.googl
e.c om");

System.out.println("Navigated to
Google");

// Wait & Navigate back

try {

    Thread.sleep(2000);

} catch (InterruptedException
e) {

    e.printStackTrace();

}

driver.navigate().back();

System.out.println("Navigated Back");

// Wait & Navigate forward

driver.navigate().forward();

System.out.println("Navigated
Forward");

```

```

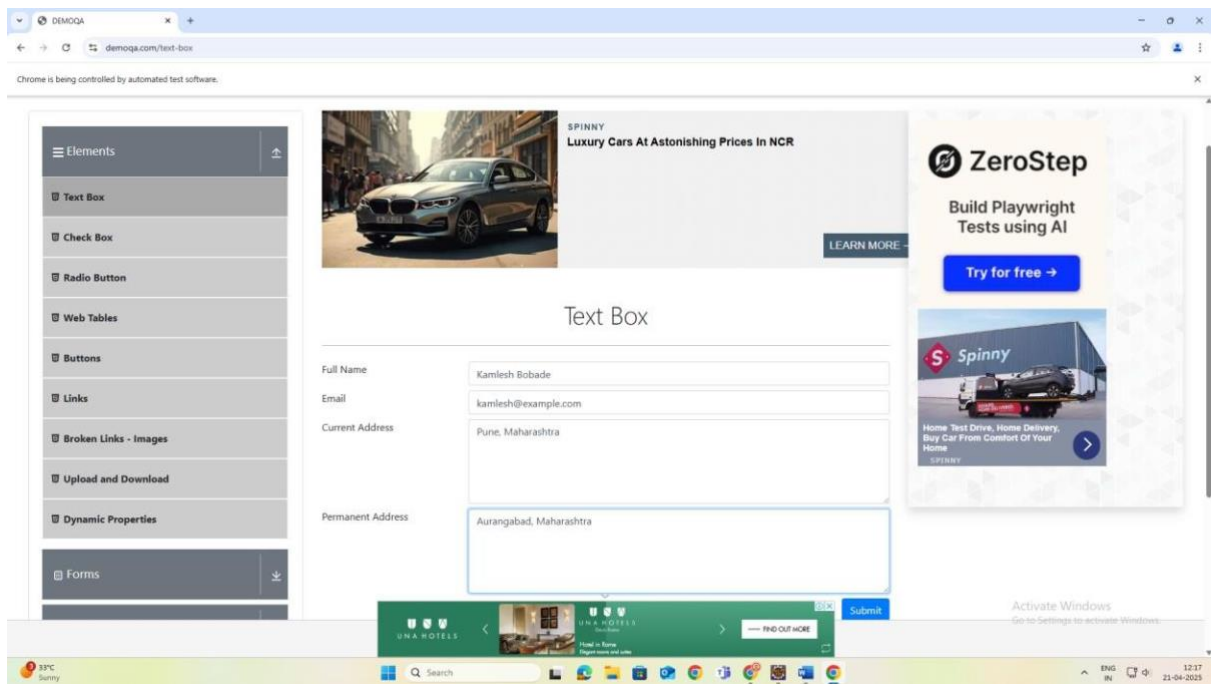
// Refresh
driver.navigate().refresh();

System.out.println("Page Refreshed");

}

// Close the browser
driver.quit();
}

```



4 Write a Java Selenium script that to automate for Handling Web Elements:to Automate web Locators using css selectors methods.

Code : package

Programs;

```
import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement;
```

```
import
org.openqa.selenium.chrome.ChromeDriver; public class Program4 { public
```

```
static void main(String[] args) {
```

```
    // Set ChromeDriver path
```

```
System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

```
    // Launch browser
```

```
    WebDriver driver = new
ChromeDriver();
driver.manage().window().maximize();
```

```
    // Step 1: Open saucedemo.com
```

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

```
        System.out.println(" Waiting 10
seconds on login page...");
```

```
        try {
```

```
            Thread.sleep(10000); // 10 seconds
        } catch (InterruptedException e) {
```

```
            e.printStackTrace();
```

```
        }
```

```
        // Step 2: Login using CSS Selectors
```

```
        WebElement username =
driver.findElement(By.cssSelector("input#u
ser-name"));
```

```
        WebElement password =
driver.findElement(By.cssSelector("input#p
assword"));
```

```
        WebElement loginBtn =
driver.findElement(By.cssSelector("input#l
ogin-button"));
```

```
username.sendKeys("standard_user");
```

```
password.sendKeys("secret_sauce");
```

```
loginBtn.click();
```

```
        // Step 3: Wait 1 minute after login
```



```
System.out.println(" Logged in
successfully.");
```

```
System.out.println(" Waiting 1
minute before closing...");
```

```
e.printStackTrace();
}
```

```
// Step 4: Close browser
```

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

```
try {
```

```
Thread.sleep(60000); // 60 seconds
```

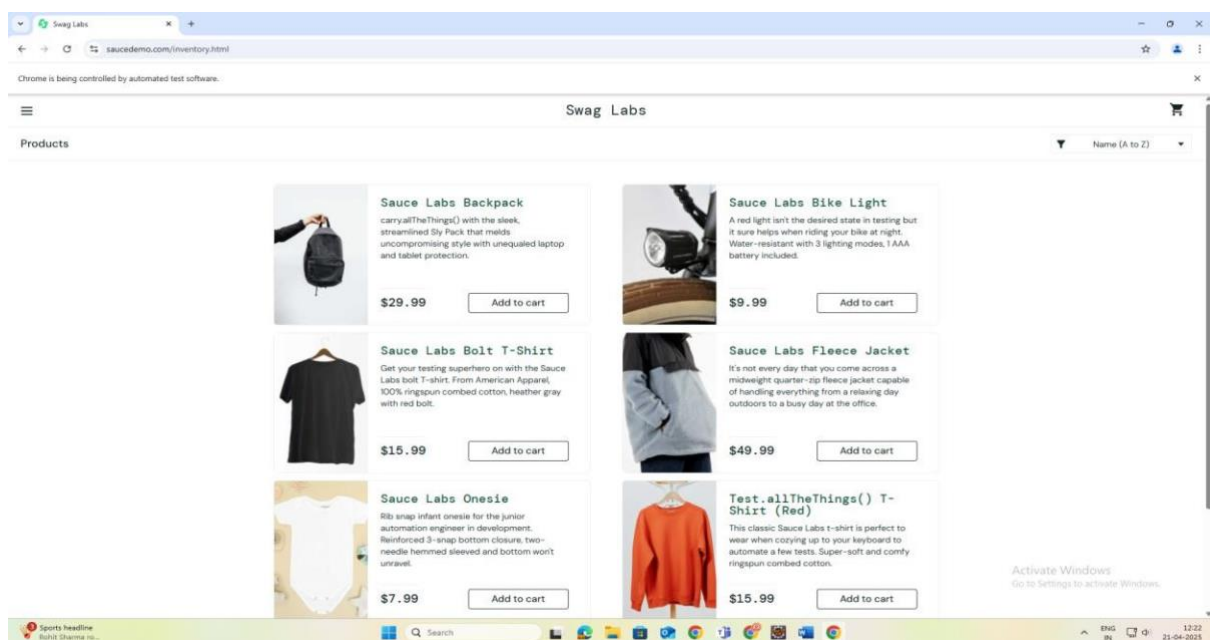
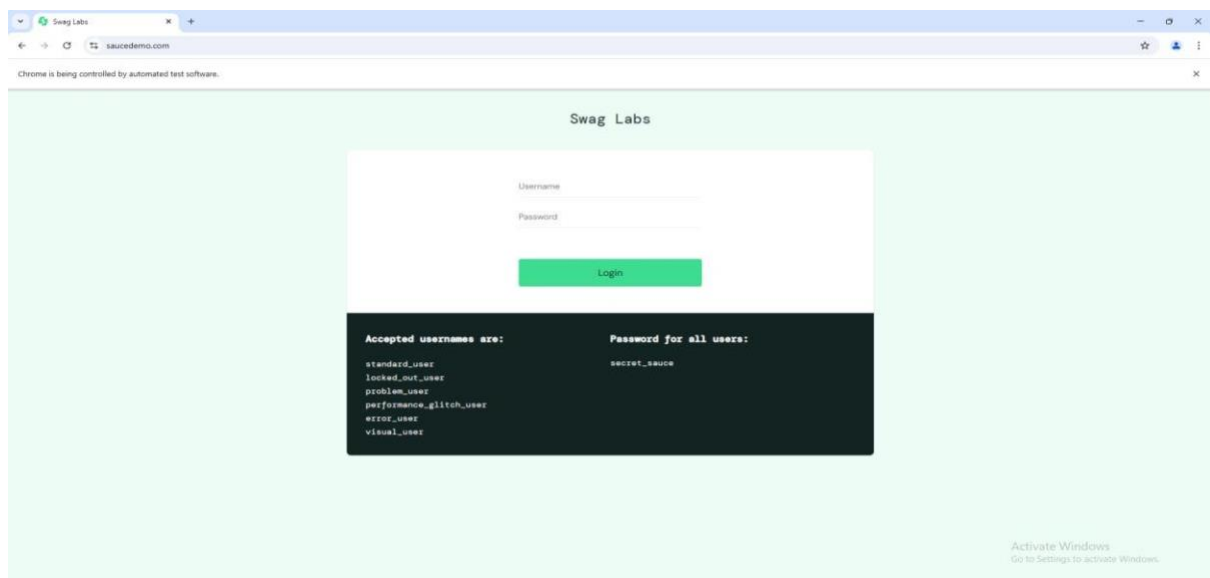
```
= 1 minute
```

```
} catch (InterruptedException e) {
```

```
System.out.println(" Browser
closed.");
```

```
}
```

```
}
```



5 Write a Java Selenium script that to automate for Handling Web Elements:to Automate web driver wait methods with multiple browser.

Code :

```
package Programs;

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.support.ui.Expected
Conditions;

import
org.openqa.selenium.support.ui.WebDriver
Wait; import java.time.Duration;

public class Chrome { public static
void main(String[] args) {

    // Set ChromeDriver path

    System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");

    // Launch Chrome browser

    WebDriver driver = new
ChromeDriver();
    driver.manage().window().maximize(); //
Open Amazon login page

    driver.get("https://www.amazon.in/ap/signin");

    // Set explicit wait (10 seconds)

    WebDriverWait wait = new
WebDriverWait(driver,
Duration.ofSeconds(10));

    // Wait for the email input field to be
visible and clickable

    WebElement emailField =
wait.until(ExpectedConditions.elementToBe
eClickable(By.id("ap_email")));

    emailField.sendKeys("your_email@example.com"); // Enter your email

    // Wait for the continue button to be
visible and clickable

    WebElement continueButton =
wait.until(ExpectedConditions.elementToBe
eClickable(By.id("continue")));
    continueButton.click();
```

```

    }

    // Wait for 1 minute (60 seconds) on
    the Amazon login page after entering
    email

    // Close the browser after 1 minute

    System.out.println(" Waiting 1
    minute on Amazon login page...");

    driver.quit();

    System.out.println(" Browser closed
    after waiting 1 minute.");
}

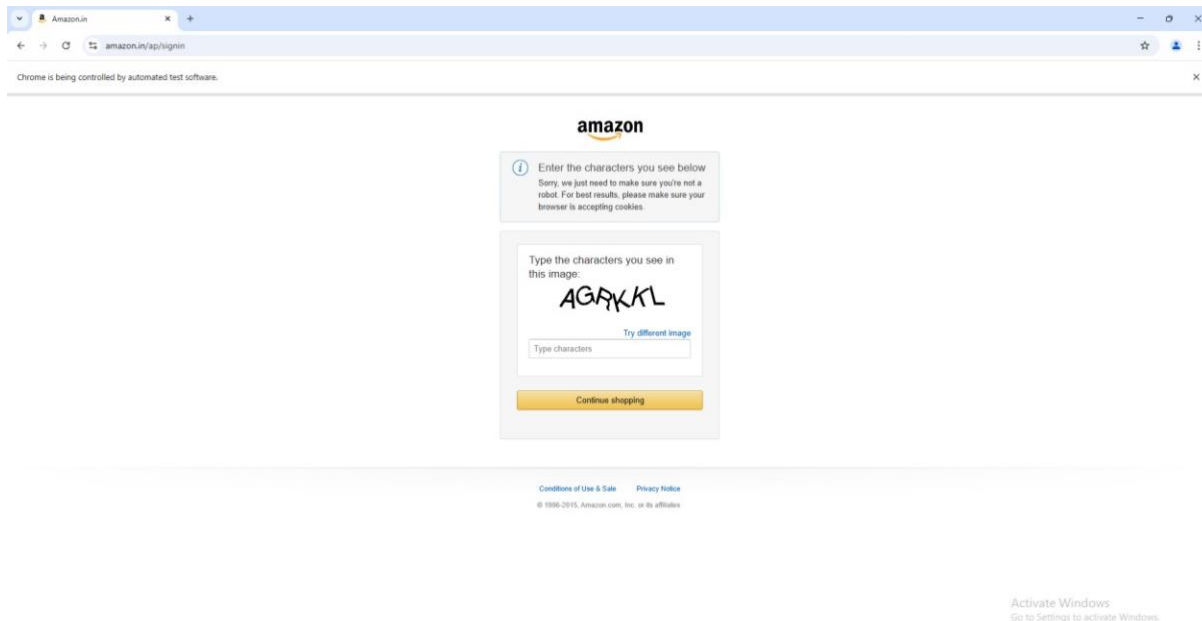
Thread.sleep(60000); // Wait for 1
minute (60,000 milliseconds)

} catch (InterruptedException e) {

}

e.printStackTrace();

```



6 Write a Java Selenium script that to automate for Handling Web Elements: Automate checkbox and confirmation buttons and prompt alert messages.

Code :

package Programs;

```

import org.openqa.selenium.Alert;

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.support.ui.Expected
Conditions;

import
org.openqa.selenium.support.ui.WebDriver
Wait; import java.time.Duration;

```

```

public class CheckBox{    public static

```

```

void main(String[] args) {

```

```

    // Set ChromeDriver path

```

```

System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");

```

```

    // Launch Chrome browser

```

```

    WebDriver driver = new
ChromeDriver();
driver.manage().window().maximize();

```

```

    // Open the "The Internet" website
driver.get("https://the-
internet.herokuapp.com/");

```

```

// Set WebDriverWait

```

```

    WebDriverWait wait = new
WebDriverWait(driver,
Duration.ofSeconds(10));

```

```

// 1. Handle Checkboxes

```

```

    System.out.println(" Handling
Checkboxes...");

```

```

    WebElement checkboxLink =
wait.until(ExpectedConditions.elementToBe
Clickable(By.linkText("Checkboxes")));

```

```

    checkboxLink.click(); // Navigate to
the checkboxes page

```

```

    // Wait for checkboxes to be visible
and interact with them

```

```

    WebElement checkbox1 =
wait.until(ExpectedConditions.elementToBe
Clickable(By.xpath("//input[@type='checkbox']
[1]")));

```

```

    checkbox1.click(); // Check the first
checkbox

```

```

    WebElement checkbox2 =
wait.until(ExpectedConditions.elementToBe
Clickable(By.xpath("//input[@type='checkbox']
[2]")));

```

```

    checkbox2.click(); // Check the
second checkbox

```

```

    // Optional: Uncheck the second
checkbox

```

```

    checkbox2.click(); // Uncheck the
second checkbox

```

```
// 2. Handle Confirmation Button  
(Alert)
```

```
System.out.println(" Handling  
Confirmation Alert...");
```

```
WebElement confirmationLink =  
wait.until(ExpectedConditions.elementToBe  
eClickable(By.linkText("JavaScript  
Alerts")));
```

```
confirmationLink.click(); // Navigate  
to the JavaScript alerts page
```

```
WebElement confirmButton =  
wait.until(ExpectedConditions.elementToBe  
eClickable(By.xpath("//button[text()='Click  
for JS Confirm']")));
```

```
confirmButton.click(); // Click the  
confirmation button
```

```
// Wait for the confirmation alert and  
accept it
```

```
Alert confirmationAlert =  
driver.switchTo().alert();
```

```
System.out.println("Alert text: " +  
confirmationAlert.getText()); // Print alert  
text
```

```
confirmationAlert.accept(); // Accept  
the confirmation alert
```

```
// 3. Handle Prompt Alert (Input text  
into the prompt alert)
```

```
System.out.println(" Handling  
Prompt Alert...");
```

```
WebElement promptButton =  
wait.until(ExpectedConditions.elementToBe  
eClickable(By.xpath("//button[text()='Click  
for JS Prompt']")));
```

```
promptButton.click(); // Click the  
prompt button
```

```
// Wait for the prompt alert to appear,  
then send input text
```

```
Alert promptAlert =  
driver.switchTo().alert();
```

```
promptAlert.sendKeys("Hello,  
Selenium!"); // Send text to the prompt
```

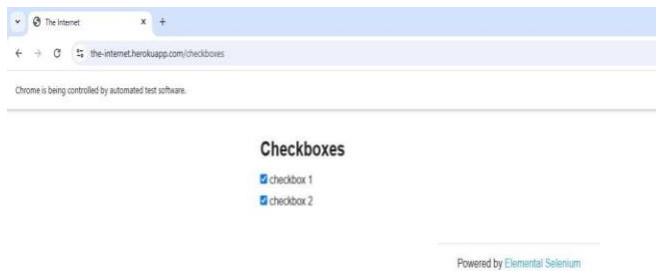
```
promptAlert.accept(); // Accept the  
prompt alert
```

```
// Close the browser  
driver.quit();
```

```
System.out.println(" Browser  
closed after handling checkboxes and  
alerts.");
```

```
}
```

```
}
```



7 Write a Java Selenium script that to automate for Handling Web Elements: Automate dropdown selection, radio button.

// Set ChromeDriver path

Code : **package**

Programs;

```
System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

import org.openqa.selenium.By; **import**

org.openqa.selenium.WebDriver;

// Launch Chrome browser

import org.openqa.selenium.WebElement;

WebDriver driver = **new**

import

org.openqa.selenium.chrome.ChromeDriver;

ChromeDriver();

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

import

org.openqa.selenium.support.ui.Select;

// Open the "DemoQA" website

import

org.openqa.selenium.support.ui.WebDriverWait; **import** java.time.Duration;

driver.get("https://demoqa.com/selectmenu");

// Set WebDriverWait

public class Alert {

WebDriverWait wait = **new**

public static void main(String[] args) {

WebDriverWait(driver,

Duration.ofSeconds(10));

```

// 1. Handle Dropdown Selection

System.out.println(" Handling
Dropdown Selection...");

WebElement dropdown =
driver.findElement(By.id("cars"));

// Use Select class to interact with the
dropdown

Select selectDropdown = new
Select(dropdown);

// Select the option by visible text
(e.g., "Volvo")

selectDropdown.selectByVisibleText("Volvo");

// Wait for 5 seconds

try {
    Thread.sleep(5000); // 5 seconds
delay
} catch (InterruptedException e) {

    e.printStackTrace();

}

```

```

// 2. Handle Radio Button Selection

System.out.println(" Handling
Radio Button Selection...");

```

```

WebElement radioButton1 =
driver.findElement(By.xpath("//label[@for
='impressiveRadio']"));

radioButton1.click(); // Select the
"Impressive" radio button

// Wait for 5 seconds

try {
    Thread.sleep(5000); // 5 seconds
delay
} catch (InterruptedException e) {

    e.printStackTrace();

}

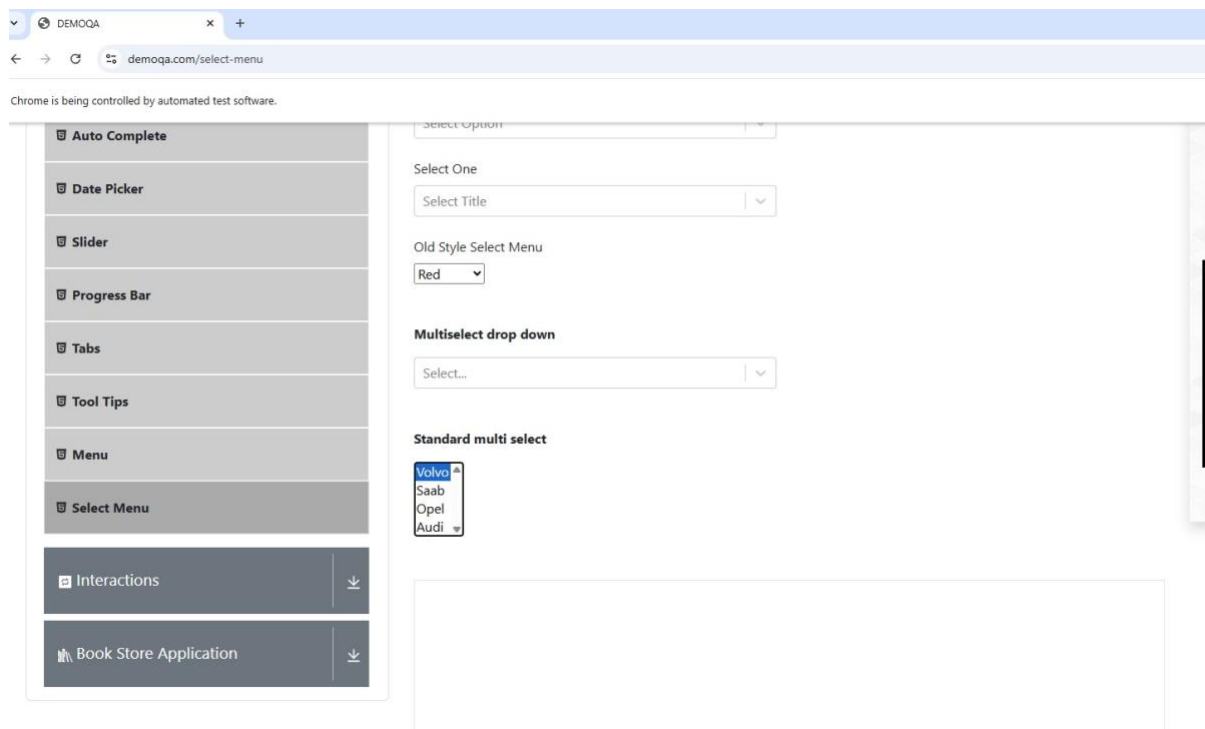
// Close the browser

driver.quit();

System.out.println(" Browser closed
after handling dropdown and radio
buttons.");

}
}

```



8 Write a Java Selenium script that to automate for Handling frames,Iframes and Nested Frames.

Code : **package**

Programs;

```
import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import
org.openqa.selenium.chrome.ChromeDriver;
```

```
public class Program5 { public static
void main(String[] args) { // Setup
ChromeDriver path
```

```
System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

```
// Create WebDriver instance
WebDriver driver = new
ChromeDriver();
driver.manage().window().maximize();
// Open website with iframe
```

```
driver.get("https://demo.guru99.com/test
/guru99home/");

try
{
```



```

// Wait 5 seconds for page to load
Thread.sleep(5000);

// Locate the iframe (YouTube video
embedded)

WebElement iframe =
driver.findElement(By.xpath("//iframe[contains(@src, 'youtube')]"));

// Switch to iframe
driver.switchTo().frame(iframe);

System.out.println(" Switched to
iframe with YouTube video.");

// Wait 5 seconds to observe the
iframe content
Thread.sleep(5000);

// You can interact with the video
here, but due to restrictions,

// we'll just print the title of the
iframe page

String iframeTitle = driver.getTitle();

System.out.println("Iframe Title: " +
iframeTitle);

// Switch back to the main page
driver.switchTo().defaultContent();

System.out.println(" Switched
back to main content.");

```

```

// Wait 5 seconds to observe the
switch

Thread.sleep(5000);

} catch (Exception e) {

    System.out.println(" Exception
occurred: " + e.getMessage());

} finally {

    // Close browser after total 15+
seconds

    driver.quit();

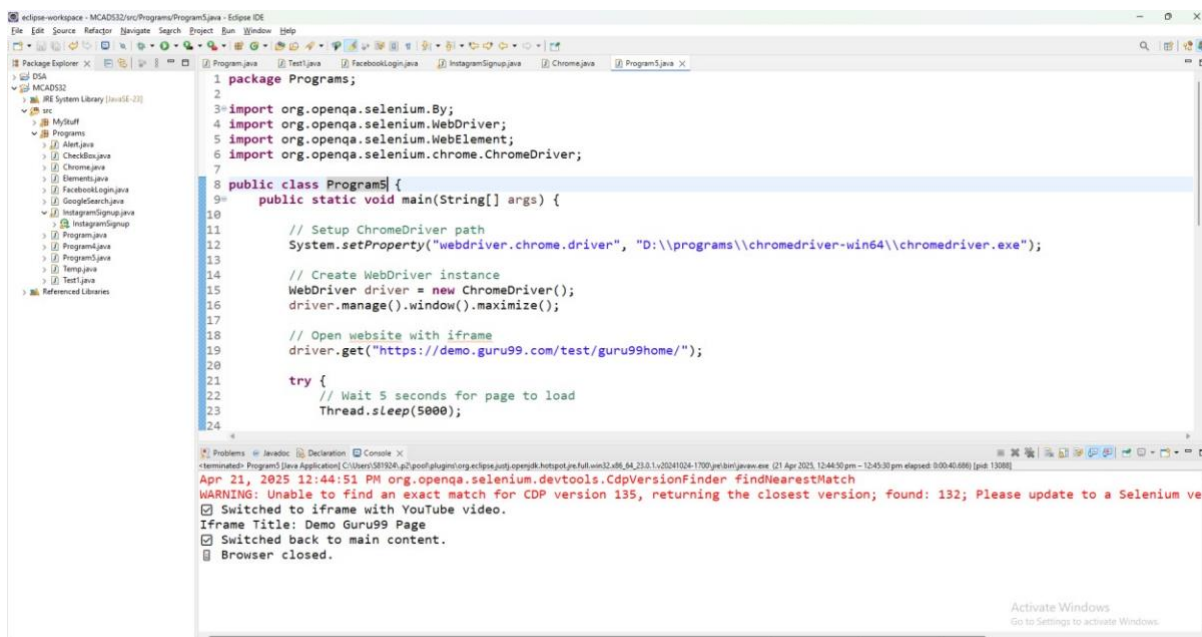
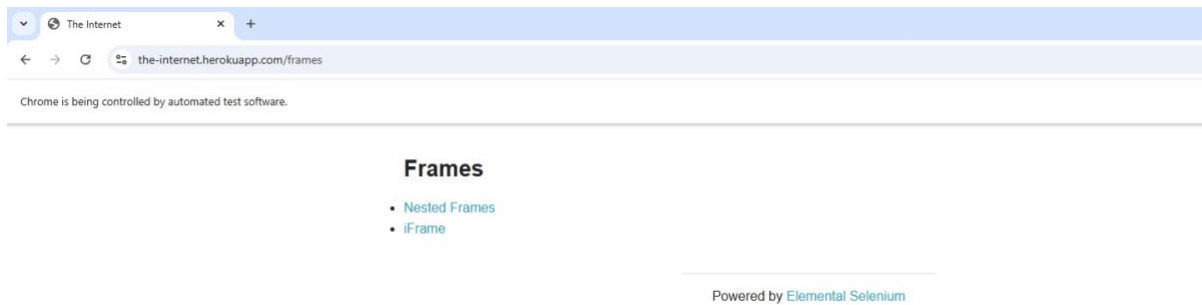
    System.out.println(" Browser
closed.");

}

}

}

```



9 Write a Java Selenium script that to automate for Alerts and Popups: Interact with Selenium alerts (simple, confirmation, and prompt).

Code :

```
package Programs;

import org.openqa.selenium.Alert;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;

// Set ChromeDriver path
public class Task {
    public static void
    main(String[] args) {
        System.setProperty("webdriver.chrome
```

```

.driver",
"D:\\programs\\chromedriverwin64\\c
hromedriver.exe");

// Launch Chrome browser
WebDriver driver = new
ChromeDriver();
driver.manage().window().maximize();

// Navigate to Alerts demo page

driver.get("https://demoqa.com/alerts");

try
{
    // Wait before actions
    Thread.sleep(5000);

    // 1️⃣ Simple Alert

    driver.findElement(By.id("alertButton")).cli
ck();

    Thread.sleep(2000); // wait for
alert to appear

    Alert simpleAlert =
driver.switchTo().alert();

    System.out.println("Simple Alert
Text: " + simpleAlert.getText());
    simpleAlert.accept(); // Click OK

    Thread.sleep(2000);

```

```

// 2️⃣ Confirmation Alert

driver.findElement(By.id("confirmButton"))
.click();

    Thread.sleep(2000);

    Alert confirmAlert =
driver.switchTo().alert();

    System.out.println("Confirmation
Alert Text: " + confirmAlert.getText());

    confirmAlert.dismiss(); // Click
Cancel

    Thread.sleep(2000);

// 3️⃣ Prompt Alert

driver.findElement(By.id("promtButton")).
click();

    Thread.sleep(2000);

    Alert promptAlert =
driver.switchTo().alert();

    System.out.println("Prompt Alert
Text: " + promptAlert.getText());
    promptAlert.sendKeys("Kamlesh");
    Thread.sleep(1000);
    promptAlert.accept(); // Click OK

    Thread.sleep(2000);

} catch (Exception e) {

    System.out.println(" Exception:
" + e.getMessage());

} finally {

```

```

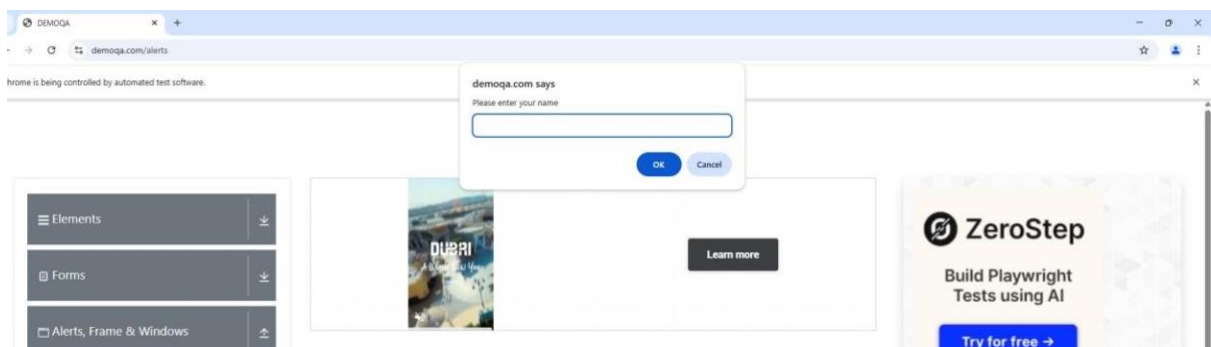
// Close browser
}

driver.quit();
}

System.out.println(" Browser closed.");
}

```

X



10 Write a Java Selenium script that automate for Submission: Automate form filling and submission.

Code :

```

package Programs;

import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement;

import
org.openqa.selenium.chrome.ChromeDriver;

public class Form {    public static void
main(String[] args) {

    // Set the path to ChromeDriver

System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");

    // Launch Chrome browser

    WebDriver driver = new
ChromeDriver();
driver.manage().window().maximize();

    // Open the form website

driver.get("https://www.techlistic.com/p/selenium-practice-form.html");

    try
{
// Wait

```

```

for
page to
load

Thread.sleep(5000);

// Fill first name
WebElement fname =
driver.findElement(By.name("firstname"));
fname.sendKeys("Kamlesh");

Thread.sleep(5000);

// Fill last name
WebElement lname =
driver.findElement(By.name("lastname"));
lname.sendKeys("Bobade");

Thread.sleep(5000);

// Select Gender radio button

driver.findElement(By.id("sex0")).click();

Thread.sleep(5000);

// Select Years of Experience

driver.findElement(By.id("exp3")).click();

Thread.sleep(5000);

```

```
// Fill Date
driver.findElement(By.id("datepicker")).sendKeys("04/21/2025");
```

```
Thread.sleep(5000);
```

```
// Select Profession checkbox
```

```
driver.findElement(By.id("profession1")).click();
```

```
Thread.sleep(5000);
```

```
// Select Automation Tool
```

```
driver.findElement(By.id("tool2")).click();
```

```
Thread.sleep(5000);
```

```
// Select Continent from dropdown
```

```
WebElement continent =
driver.findElement(By.id("continents"));
continent.sendKeys("Europe");
```

```
Thread.sleep(5000);
```

```
// Select Selenium Command
```

```
WebElement commands =
driver.findElement(By.id("selenium_commands"));
```

```
commands.sendKeys("Browser
Commands");
```

```
Thread.sleep(5000);
```

```
// Submit the form
```

```
driver.findElement(By.id("submit")).click();
```

```
Thread.sleep(5000);
```

```
} catch (Exception e) {
```

```
System.out.println(" Exception:
" + e.getMessage());
```

```
} finally {
```

```
// Close the browser
```

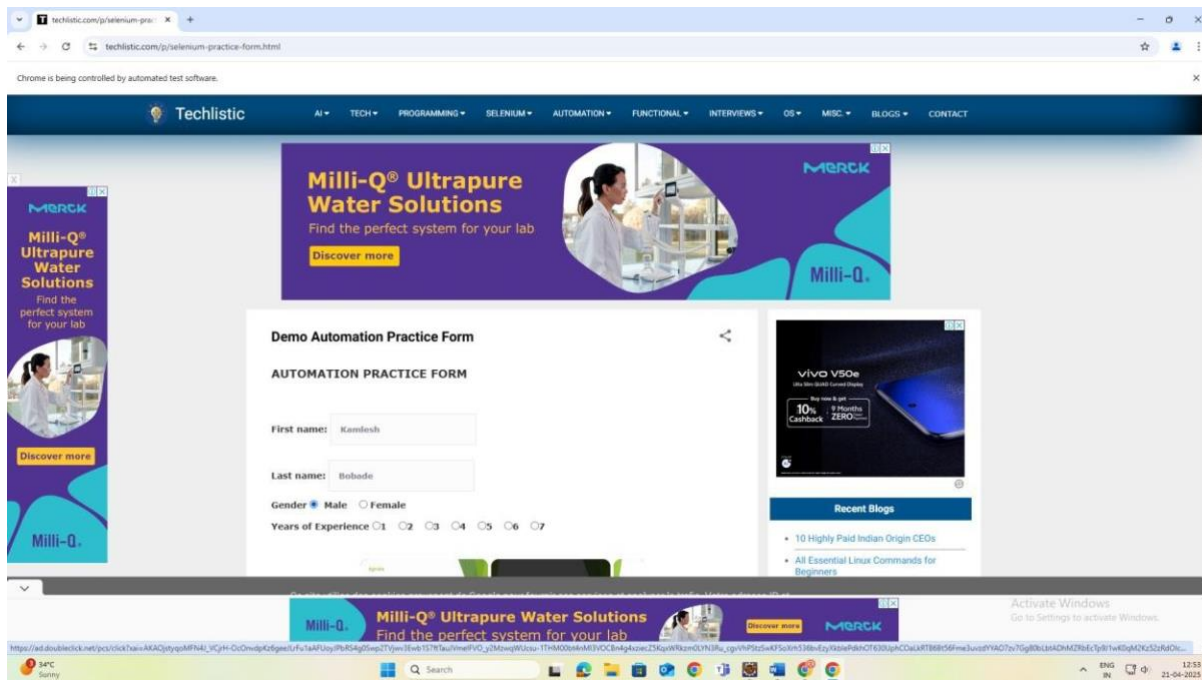
```
driver.quit();
```

```
System.out.println(" Form
submitted and browser closed.");
```

```
}
```

```
}
```

```
}
```



11 Write a Java Selenium script that automate for Browser Compatibility Testing: Execute the same test across different browsers.

```
public static void runTest(WebDriver
driver, String browserName) {
```

Code : **package**

```
try {
```

Programs;

```
System.out.println("🔧 Running
test on: " + browserName);
```

```
import org.openqa.selenium.WebDriver;
```

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

```
import
```

```
org.openqa.selenium.chrome.ChromeDriv
er;
```

```
driver.get("https://www.amazon.in");
```

```
import
```

```
org.openqa.selenium.firefox.FirefoxDriver;
```

```
Thread.sleep(5000); // Wait for 5
seconds
```

```
// import
```

```
org.openqa.selenium.edge.EdgeDriver;
```

```
String title = driver.getTitle();
```

```
public class BrowserTest {
```

```
System.out.println "[" + browserName + "]
Page Title: " + title);    } catch (Exception
e) {
```

```
        System.out.println("  Error on " +  
browserName + ": " + e.getMessage());
```

```
    } finally {        if
```

```
(driver != null) {
```

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

```
        System.out.println("  " +  
browserName + " browser closed.\n");
```

```
    }
```

```
}
```

```
}
```

```
public static void main(String[] args) {
```

```
    // CHROME
```

```
    try {
```

```
System.setProperty("webdriver.chrome.dri  
ver",  
"D:\\programs\\chromedriverwin64\\chro  
medriver.exe");
```

```
        WebDriver chromeDriver = new  
ChromeDriver();  
runTest(chromeDriver, "Chrome");    }
```

```
catch (Exception e) {
```

```
        System.out.println("  Chrome test  
failed: " + e.getMessage());
```

```
    }
```

```
    // FIREFOX
```

```
    try {
```

```
System.setProperty("webdriver.gecko.driv  
er", "D:\\programs\\geckodriver.exe");
```

```
        WebDriver firefoxDriver = new
```

```
FirefoxDriver();
```

```
runTest(firefoxDriver, "Firefox");    }
```

```
catch (Exception e) {
```

```
        System.out.println("  Skipping  
Firefox: " + e.getMessage());
```

```
    }
```

```
    // EDGE (Optional)
```

```
    /*
```

```
try {
```

```
System.setProperty("webdriver.edge.drive  
r", "D:\\programs\\msedgedriver.exe");
```

```
        WebDriver edgeDriver = new  
EdgeDriver();  
runTest(edgeDriver, "Edge");    }  
catch (Exception e) {
```

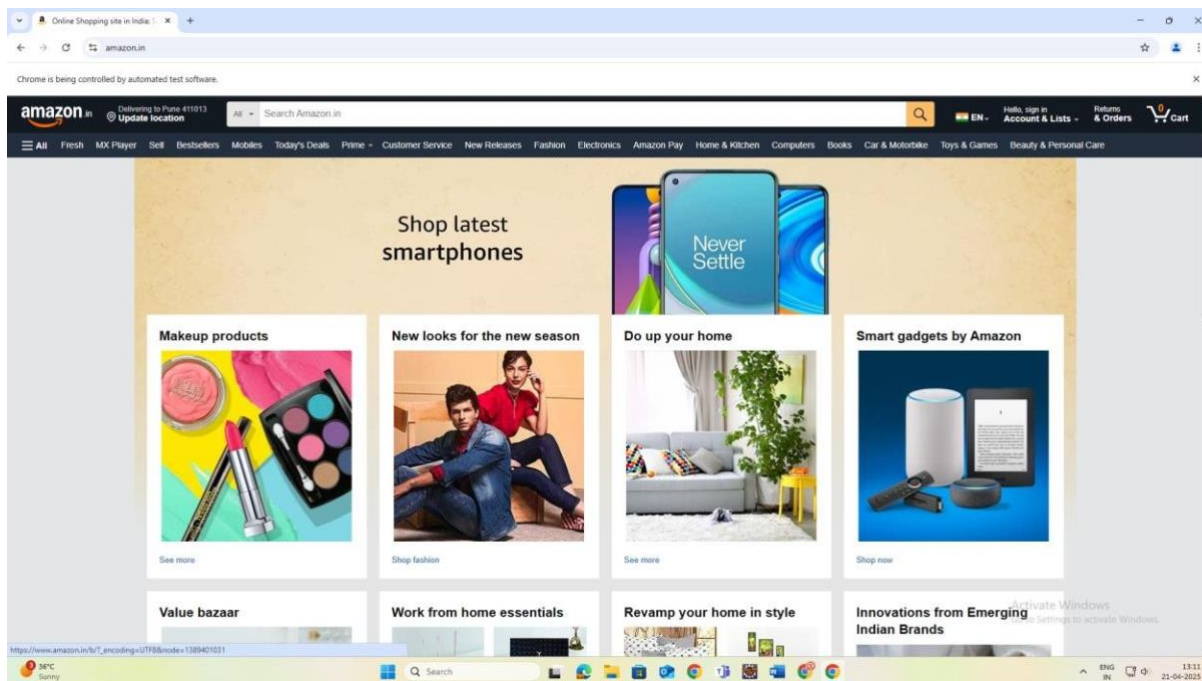
```
        System.out.println("  Skipping  
Edge: " + e.getMessage());
```

```
    }
```

```
    */
```

```
}
```

```
}
```

12 Write a Java Selenium script that automate for Drag and Drop WebElements.

Code :

// Set up ChromeDriver

```
package Programs;
```

```
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.interactions.Actions;
```

```
public class DragAndDropDemo {
```

```
    public static void main(String[] args) {
```

```
        System.setProperty("webdriver.chrome.driver",
            "D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

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

```
        try {
```

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

```
            driver.get("https://demo.guru99.com/test/drag_drop.html");
```

```
            Thread.sleep(5000); // Wait for the
            page to load
```

```

// Source element: BANK

WebElement fromBank =
driver.findElement(By.xpath("//a[contains(
text(),'BANK')]"));

// Target element: Account section
on Debit Side

WebElement toAccountDebit =
driver.findElement(By.xpath("//ol[@id='ba
nk']/li"));

// Source element: 5000

WebElement fromAmount =
driver.findElement(By.xpath("//a[contains(
text(),'5000')]"));

// Target element: Amount section
on Debit Side

WebElement toAmountDebit =
driver.findElement(By.xpath("//ol[@id='a
mt7']/li"));

// Drag and drop using Actions

Actions act = new Actions(driver);

act.dragAndDrop(fromBank,
toAccountDebit).build().perform();

Thread.sleep(5000);

```

```

act.dragAndDrop(fromAmount,
toAccountDebit).build().perform();
Thread.sleep(5000);

System.out.println(" Drag and
drop completed successfully!");

} catch (Exception e) {

System.out.println(" Error
occurred: " + e.getMessage());

} finally {

try {

Thread.sleep(10000); // Wait for
10 seconds before closing

} catch (InterruptedException e) {

e.printStackTrace();

}

driver.quit();

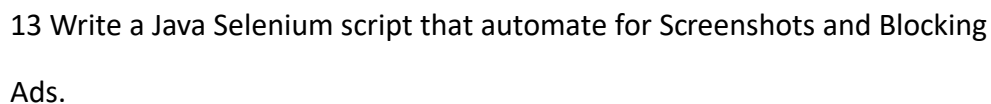
System.out.println(" Browser
closed.");

}

}

}

```



```
import java.io.File;
```

```

import java.util.concurrent.TimeUnit;

public class ScreenshotWithSeleniumSite {

    public static void main(String[] args) {

        // Set the path to ChromeDriver

        System.setProperty("webdriver.chrome.driver",
            "D:\\programs\\chromedriverwin64\\chromedriver.exe");

        // Chrome options to disable popups
        (minimal ads on this site anyway)

        ChromeOptions options = new
        ChromeOptions();

        options.addArguments("--
        disablepopup-blocking");

        // Launch browser

        WebDriver driver = new
        ChromeDriver(options);

        try

        {

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

            driver.manage().timeouts().implicitlyWait(
            10, TimeUnit.SECONDS);

            // Open Selenium official site

```

```

import java.io.IOException;

driver.get("https://www.selenium.dev/");
Thread.sleep(5000); // Wait for page to
load fully

        // Take screenshot of homepage

        TakesScreenshot screenshot =
        (TakesScreenshot) driver;

        File src =
        screenshot.getScreenshotAs(OutputType.F
        ILE);

        File dest = new
        File("D:\\programs\\selenium_homepage
        _screenshot.png");

        org.openqa.selenium.io.FileHandler.copy(s
        rc, dest);

        System.out.println(" Screenshot
        saved at: " + dest.getAbsolutePath());

        } catch (InterruptedException |
        IOException e) {

            System.out.println(" Error: " +
            e.getMessage());

        } finally {

            try {

                Thread.sleep(5000); // Delay
                before closing

            } catch (InterruptedException e) {

                e.printStackTrace();

```

```

    }

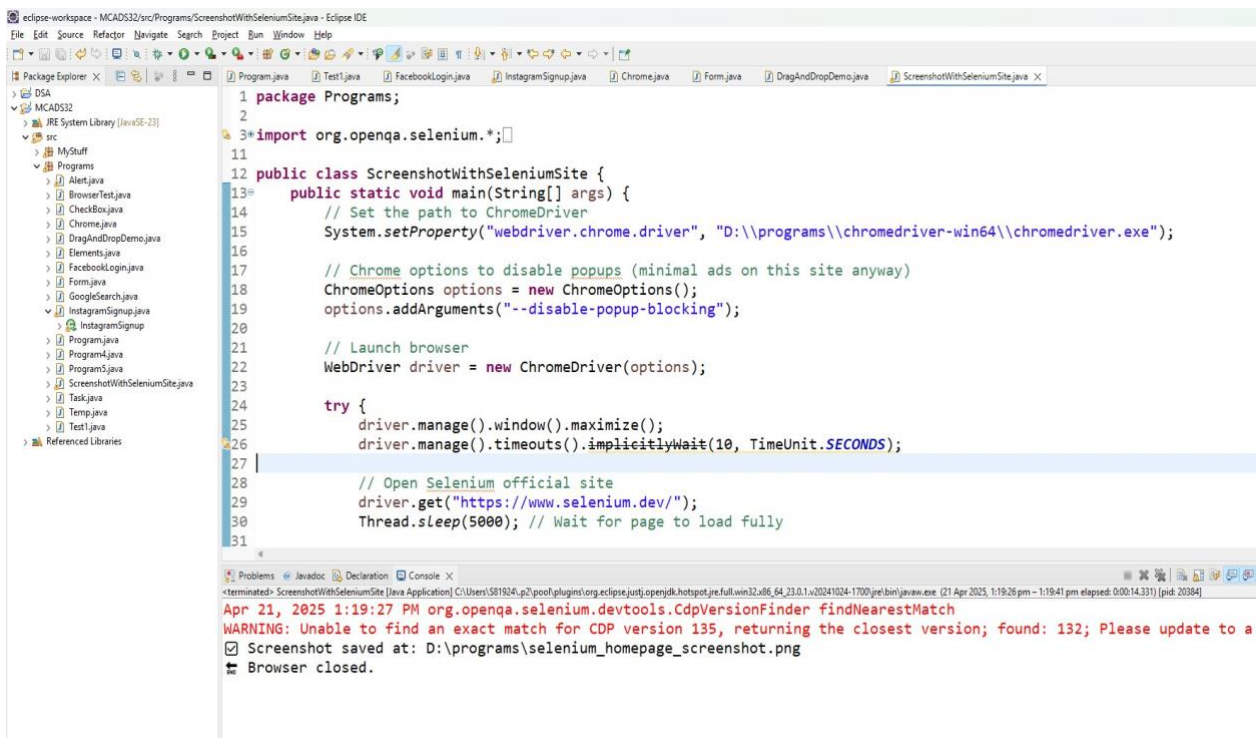
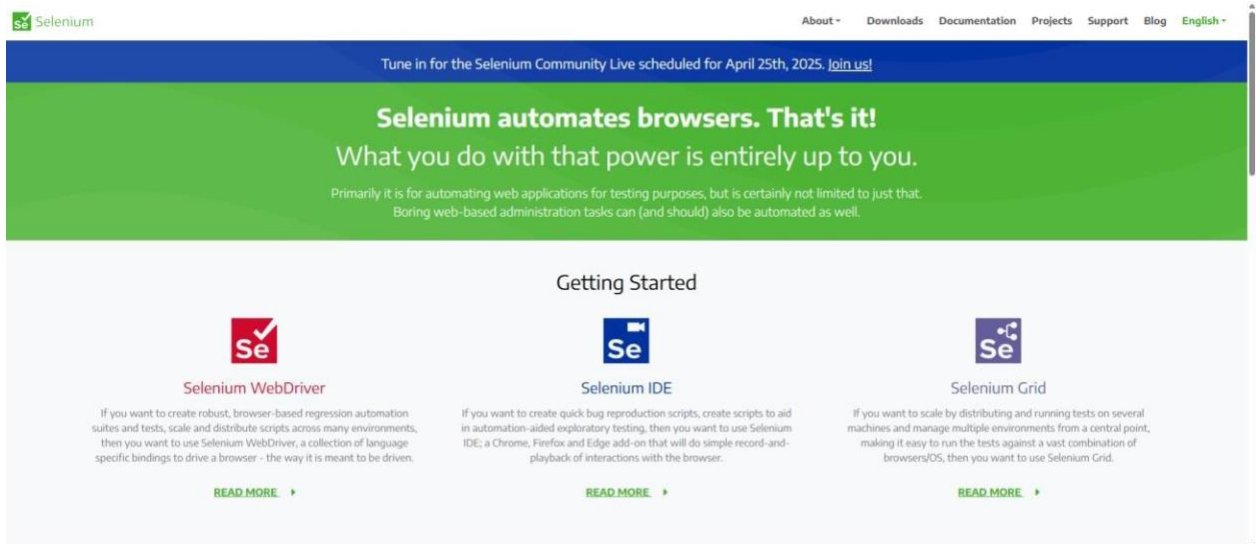
    driver.quit();

    System.out.println(" Browser    } closed.");

}

}

```



14 Write a Java Selenium script that automate to opens the web table from the url

Code : package

Programs;

```
import org.openqa.selenium.By; import
org.openqa.selenium.WebDriver; import
org.openqa.selenium.WebElement;
import
org.openqa.selenium.chrome.ChromeDriver;
```

```
import java.util.List;
```

```
public class WebTableAutomation {
public static void main(String[] args) {
```

```
    // Set the path for ChromeDriver
```

```
    System.setProperty("webdriver.chrome.driver",
"D:\\programs\\chromedriverwin64\\chromedriver.exe");
```

```
    // Launch Chrome
```

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

```
    try
{
```

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

```
    driver.get("https://demo.guru99.com/test
```

```
/web-table-element.php");
```

```
    Thread.sleep(5000); // Wait for table to
load
```

```
    // Locate the table
```

```
    WebElement table =
driver.findElement(By.className("dataTable"));
```

```
    // Get all rows
```

```
    List<WebElement> rows =
table.findElements(By.tagName("tr"));
```

```
    // Loop through each row and print
cells    for (WebElement row : rows) {
```

```
        List<WebElement> cells =
row.findElements(By.tagName("td"));
```

```
        for (WebElement cell : cells) {
            System.out.print(cell.getText()
+ "\t");
```

```
        }
```

```
        System.out.println();
```

```
    }
```

```
    } catch (Exception e) {
```

```
        System.out.println(" Error: " +
e.getMessage());
```

```
    } finally {
```

```
try {
```

```
    Thread.sleep(5000); // Keep
browser open for 5 sec before closing
```

```

    } catch (InterruptedException e) {

    }

    e.printStackTrace();

    driver.quit();

    System.out.println(" Browser
closed.");

}

```

```

1 package Programs;
2
3 import org.openqa.selenium.By;
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Problems Declaration Console

<terminated> WebTableAutomation [Java Application] C:\Users\S81924\p2\poof\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_23.0.1.v20241024-1700\jre\bin\javaw.exe (21 Apr 2025, 1:22:53 pm - 1:23:12 pm elapsed: 0:00:18.224) [pid: 127]

Apr 21, 2025 1:22:54 PM org.openqa.selenium.devtools.CdpVersionFinder findNearestMatch

WARNING: Unable to find an exact match for CDP version 135, returning the closest version; found: 132; Please u

Company	Group	Prev Close (Rs)	Current Price (Rs)	% Change
Kwaliti A	A	74.9	111.9	+ 6
Cera Sanitaryware L	A	118.3	647.6	+ 7.1
Rashtriya Chemicals	A	414.3	870.4	+ 2.9
YES Bank Ltd. A	A	306.6	328.6	+ 5.1
Punj. NationlBak	A	763.2	632.3	+ 4.5
Kansai Nerolac Paint	A	983.6	57.9	+ 10
ICICI Pru Life A	A	649.3	781.1	+ 7.6
Asian Paints Ltd.	A	366.3	535.2	+ 7.8
YES Bank Ltd. A	A	33.2	138.9	+ 2.7
GRUH Finance Li A	A	306.1	992.8	+ 8.8
Asian Paints Ltd.	A	455.3	177.7	+ 5.9
Chennai Petro. A	A	688.6	266.4	+ 7.5
PI Industries A	A	702.2	113.1	+ 7.3
Hero MotoCorp A	A	275.5	753.7	+ 5.6
Cera Sanitaryware L	A	224.6	791.6	+ 3.8
Bajaj Corp Ltd. A	A	649.7	69.1	+ 8
Quess Corp A	A	111.9	160.1	+ 7.6
Bharat Electroni	A	720.6	973.5	+ 6.6
YES Bank Ltd. A	A	737.6	292.2	+ 7.1
IDFC L A	A	148.4	+ 4.9	
Mangalore Refine	A	34	338.9	+ 6.1
CESC Ltd. A	A	295.9	503	+ 7.3
PI Industries A	A	139.6	546.1	+ 6
Nestle India A	A	925.1	505.7	+ 3.8
Asian Paints Ltd.	A	19.8	527.9	+ 7.2
Jagran Prakashan	A	577	362.4	+ 5.1

Browser closed.

Web Table Elements

demo.guru99.com/test/web-table-element.php

Chrome is being controlled by automated test software.

GURU99 Demo Site

Our Proud Sponsors Zoho BugTracker Get Free Nine

Selenium Insurance Project Agile Project Bank Project Security Project Telecom Project Payment Gateway Project New Tours

SEO

Top Gainers BSE | NSE Last updated: 11 Dec, 13:51

Gainers Losers

Daily Weekly Monthly

All | Group A | Group B | Group M | Group T | Group MT | Group Z

Company	Group	Prev Close (Rs)	Current Price (Rs)	% Change
Deepak Fertilisers	A	854.1	127.1	+ 6.5
Nalkar Corporation L	A	312.4	48.9	+ 4.9
Coir & Kings L	A	341.7	884.9	+ 2.9
Max Financial Servic	A	368.8	321	+ 8.2
LIC Housing FI	A	465.7	734	+ 8.5
IRB Infrastructure	A	430.4	116	+ 9.3
JagprakashAssociates	A	454.1	436.4	+ 5.1
Apollo Hospitals	A	448.2	872.9	+ 6.3
Corporation Bank	A	188	636.3	+ 6
Rashtriya Chemicals	A	249.9	187.1	+ 9.3
Power Finance Co	A	536.8	89.1	+ 5.6
NIT Technologies	A	500.7	120	+ 9.7
Bata India	A	727	456	+ 9.3
Strides Pharma Sc	A	137.9	417.1	+ 4.7
Kapra Ceramics	A	170.3	124.3	+ 5.7
Nalkar Corporation L	A	671.7	961.6	+ 7.3

Activate Windows Go to Settings to activate Windows.

9°C Sunny

Search

13:23 21-04-2025

15 Implement TestNG Framework for Automation Testing: Create test suite with TestNG.

```
Code: package selenium;

import org.openqa.selenium.By; import org.openqa.selenium.WebDriver; import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver; import org.testng.annotations.*;

public class GoogleSearchTest {

    WebDriver driver;

    @BeforeClass
    public void setup() {

        System.setProperty("webdriver.chrome.driver", "D:\\(SEM 4) MCADS\\STQA\\MySetup\\Driver\\chromedriver-win64\\chromedriver.exe");
        driver = new ChromeDriver();
        driver.manage().window().maximize();
    }

    @Test
    public void testGoogleSearch()
        throws InterruptedException {
        driver.get("https://www.google.com");

        Thread.sleep(1000);

        WebElement searchBox = driver.findElement(By.name("q"));
        searchBox.sendKeys("Selenium WebDriver");

        Thread.sleep(1000);
        searchBox.submit();

        Thread.sleep(3000);
    }

    @AfterClass
    public void tearDown() {
        driver.quit();
    }
}
```

2. testng.xml

```
<!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd" >

<suite name="AutomationSuite">

    <test name="GoogleSearchTest">
```



```

<classes>

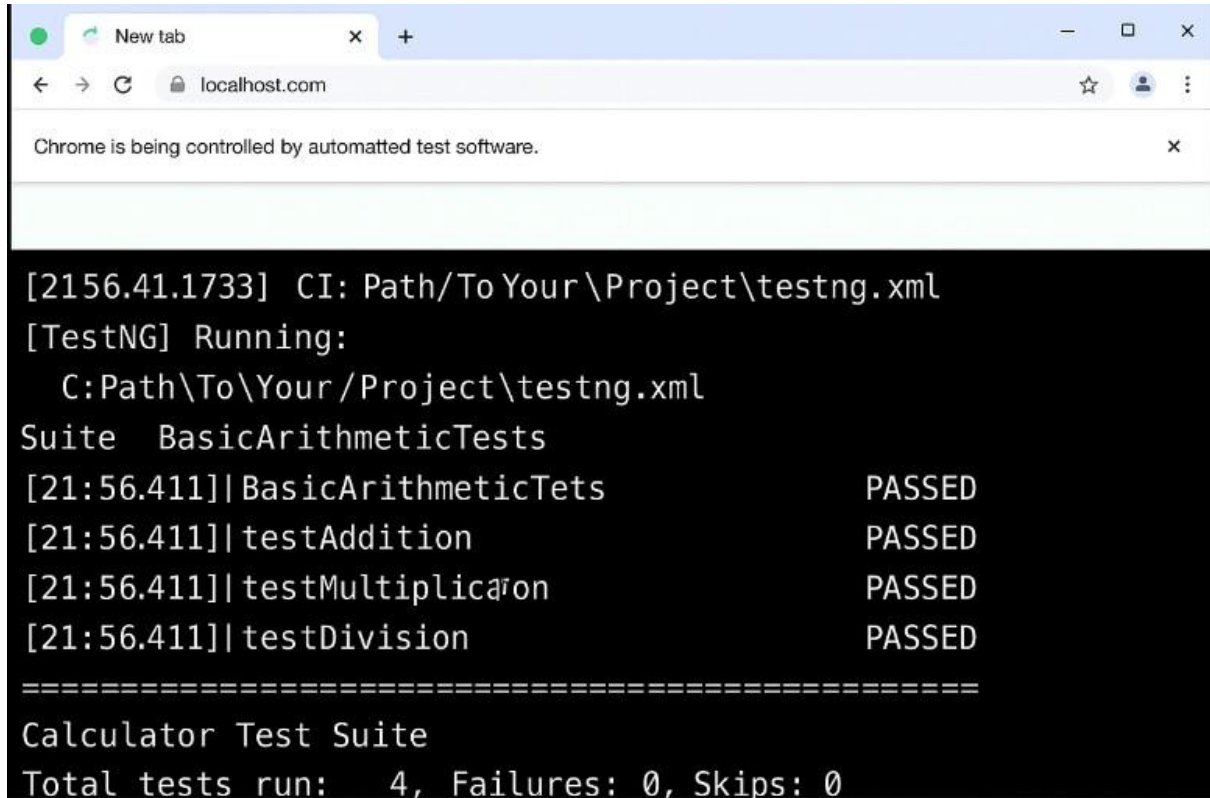
  <class
name="selenium.GoogleSearchTest"/>

</classes>

</test>

</suite>

```



```

[2156.41.1733] CI: Path/To Your \Project\testng.xml
[TestNG] Running:
  C:Path\To\Your /Project\testng.xml
Suite  BasicArithmeticTests
[21:56.411]|BasicArithmeticTets                PASSED
[21:56.411]|testAddition                        PASSED
[21:56.411]|testMultiplication                 PASSED
[21:56.411]|testDivision                      PASSED
=====
Calculator Test Suite
Total tests run:   4, Failures: 0, Skips: 0

```



```

[RemoteTestNG] detected TestNG version ...
Starting ChromeDriver ...
ChromeDriver was started successfully
Launching browser...
Test passed: testGoogleSearch
Browser closed...
PASSED: testGoogleSearch
-----
AutomationSuite
Tests run: 1, Failures: 0, Skips: 0

```