Experiment – 10

Student Name: Vivek Kumar   **UID:21BCS8129**
Branch: BE-CSE (LEET)   **Section/Group: 801 "B"**
Semester: 6ᵀᴴ   **Subject Code: 20CSP-380**
Subject Name: ST LAB

1. Aim/Overview of the practical:

Write and test a program to login in web page.

2. Objective:

A login screen is a web page or an entry page to a web/mobile application that requires user identification and authentication, regularly performed by entering a username and password combination. The login process is the most essential feature for any system/application as it provides access to an entire web-site/application or part of it. So, testing of the login screen needs complete coverage.
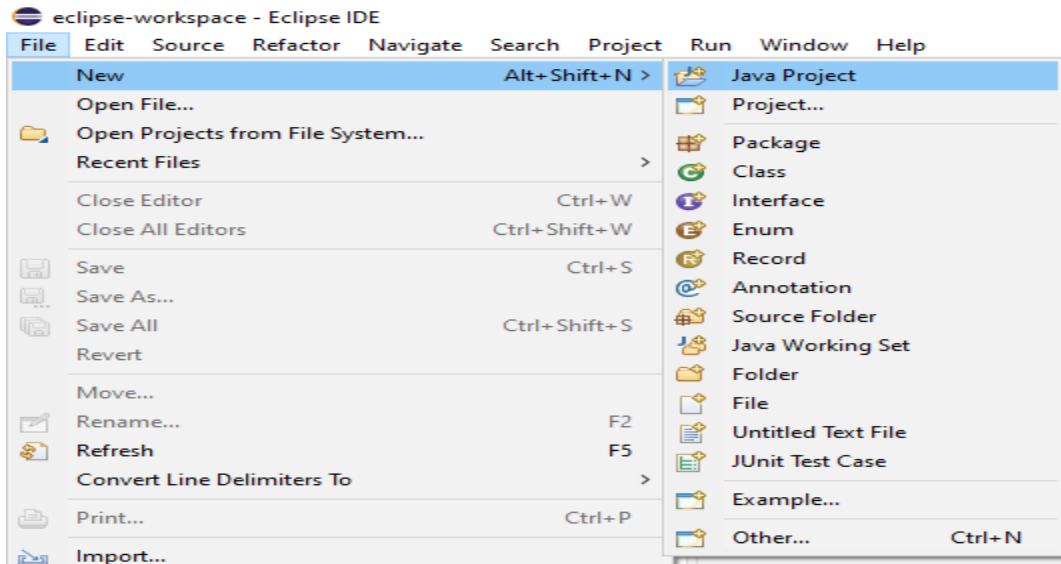
3. Requirement Analysis:

Google Chrome, Website or webpage, Online tool.

4. Hardware Requirement:

Windows/Mac OS, Computer, Window Power Supply.
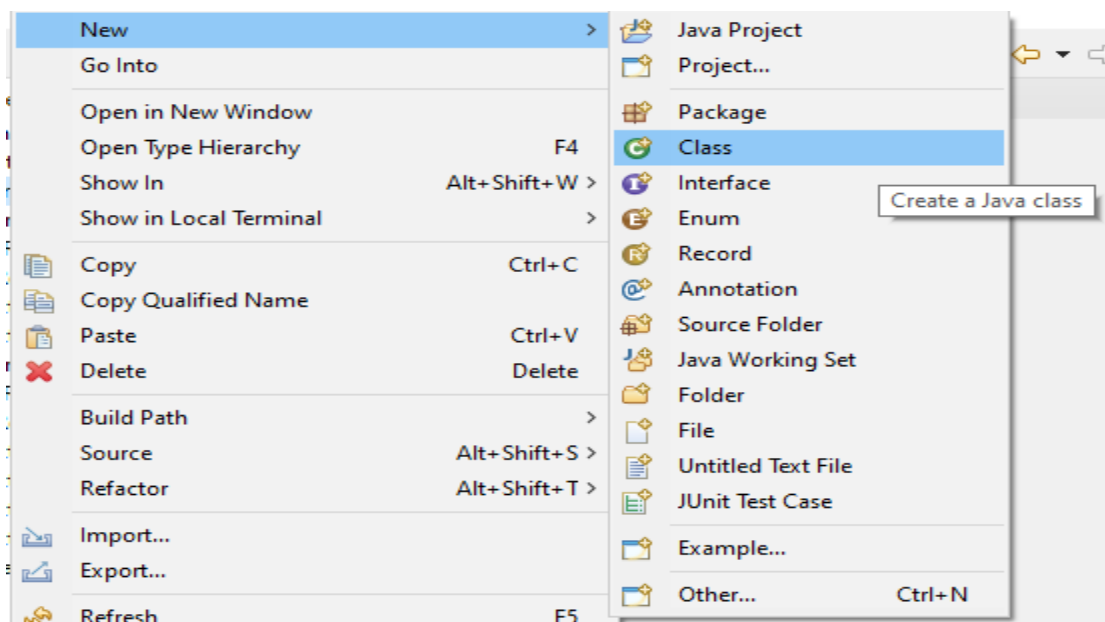
5. Procedure/ Algorithm/ Code:

Step 1: To Create New Project, Click on File –> New –> Java Project.

After creating a new project, give the project name "Login_Test" and click on Finish.

Step 2: Create a new java class

For creating a new Java class, Right-click on "Test", Click on new and select Class. Given the class name "Login_test" and click Finish.
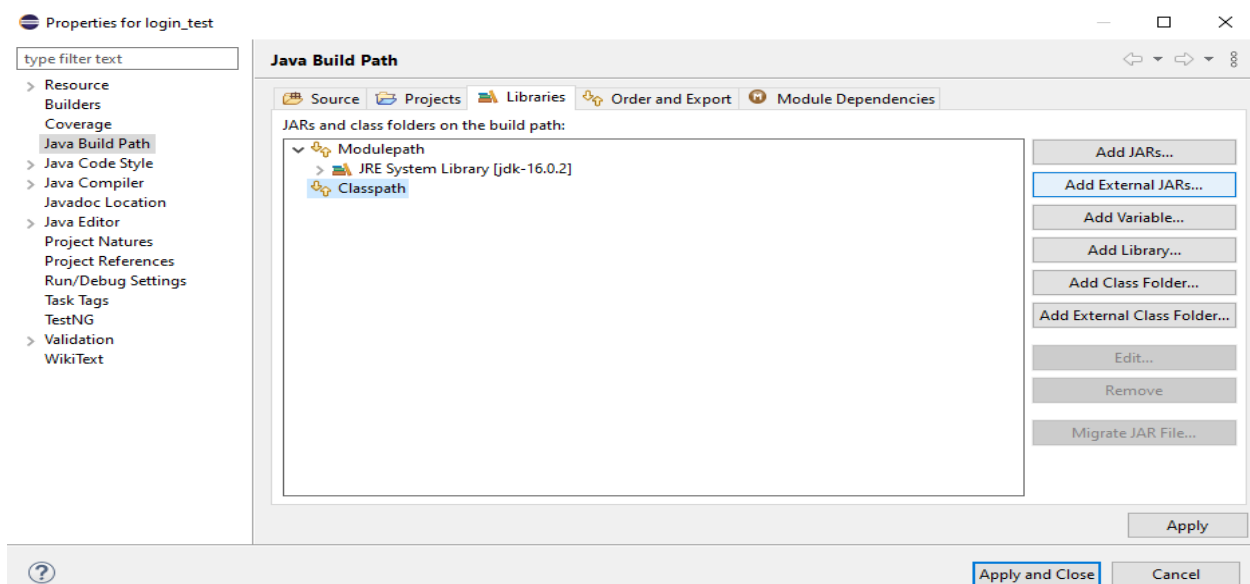
Now we have to import Selenium Web Driver and Chrome Driver.

Step 3: Download the latest Chrome Driver and Selenium Web Driver for Java. After downloading, unzip the downloaded files. Right-click on "Testing", Click on Build Path, and Select Configure Build Path.
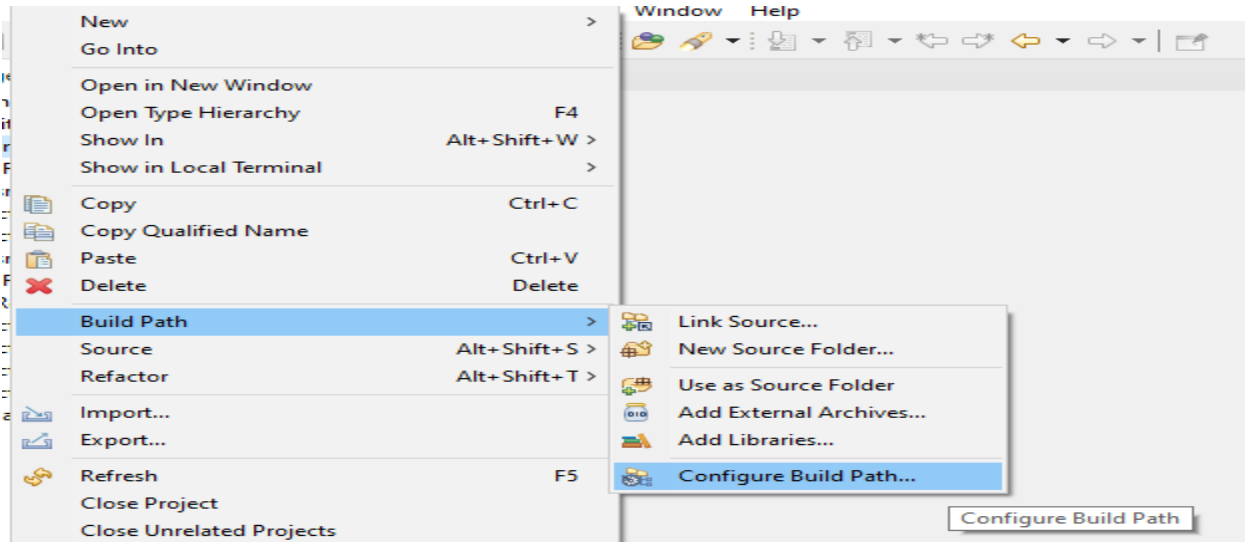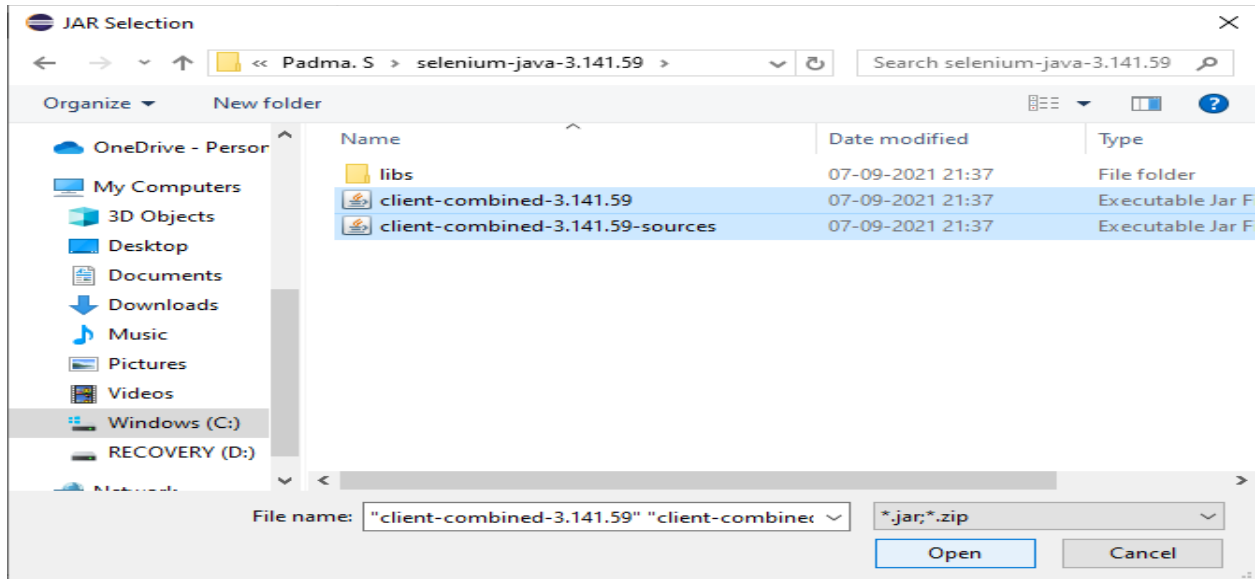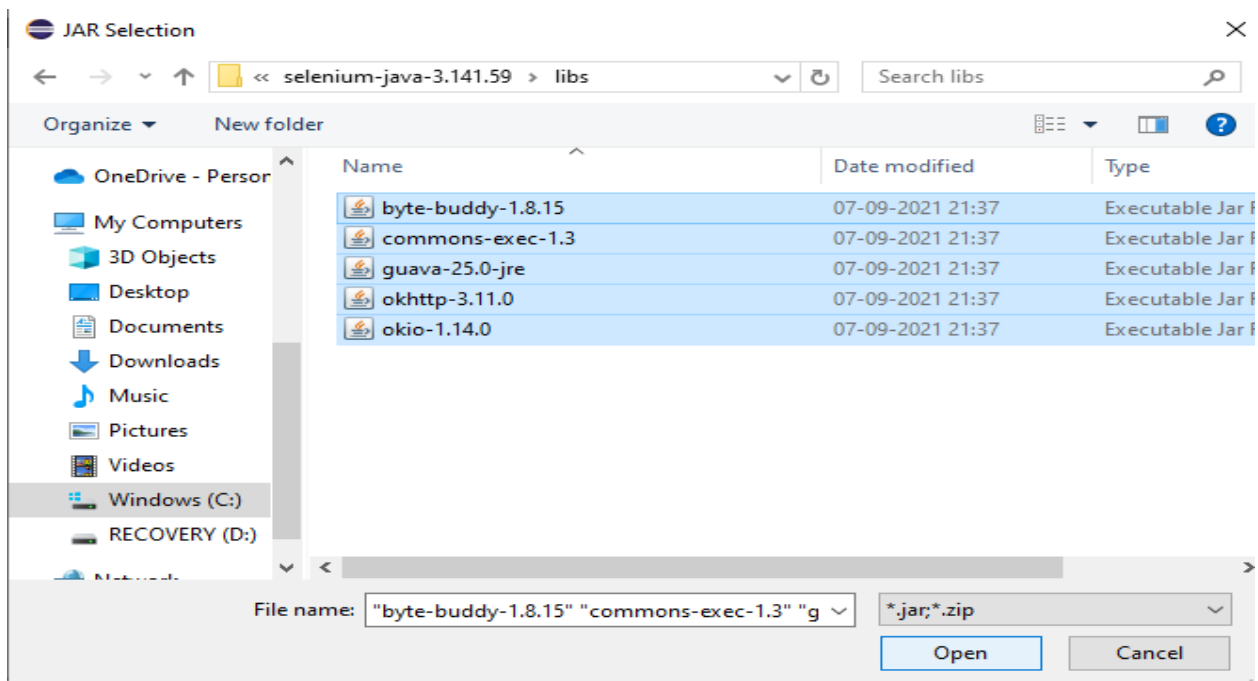

Step 4: Click on Classpath

4.1: Initially alongside select Add External JARs.

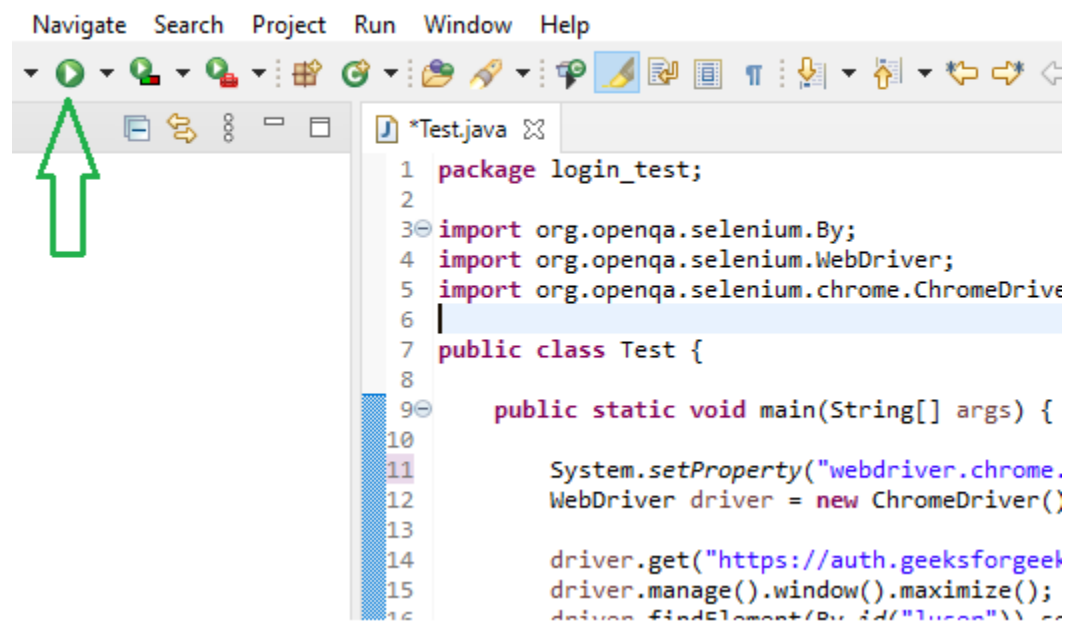4.2: Now go to the Selenium driver downloaded file, select both these jar files and click on 'Open'.

4.3: Go to the same folder, open the "libs" folder, select all the jar files, and click on 'Open'.

After adding all these required Jar files, below is the given code to test the login page.

click on the 'Run' button.

UI/UX:

·   Tab Order - Check if there is a logical order for using the tab key

·   Focus on Username field - Check if, while landing on the page, the cursor is at the username field

·   Use of enter key - Check if Login button is activated on selecting enter

·   Accessibility - Check if all the fields on the page are correctly identified and labeled

·   Look & Feel - Check if the page looks fine, and everything is aligned correctly.

·   Content - Check if the content of the page is up to the mark. Are there any typos in the labels, controls of the screen?

·   Links - Check if the page contains any existing links, and are these links still valid.

·   Responsiveness - Check the responsiveness of the login screen in multiple sizes of computer monitors.

Security Checks:

·   Password - Check if the password shown or hidden (using asterisks)

·   Password - Check if you can copy & paste the password from other applications.

·   Password - Check if there is a minimum complexity on the password

·   Password - Check if there is a 'Show password' option that is there or not. If yes, then check if it is working fine.

·   Common Password Lookup - Check if the login screen is performing a lookup in the list of the most common passwords (Common Passwords List)

·   View Source - Check the source code of the application and check if any valuable information given away in the HTML source code

·   SQL Injection - Check if the login page is vulnerable to SQL input

·   Pages - Check if you can access the other pages of the application without logging in.

·   URL Manipulation - Check if you are able to access the other pages of the application by editing the URLs, to gain access where it should not be allowed (without login).

·   Multiple accounts - Check if by using different accounts, you can be logged in at the same time in the same browser

· Cookies - Check if you can edit and/or disable the cookies.

Functionality:

· Login - Check the login functionality with valid/invalid credentials and without providing credentials.

· Logout - Check the logout functionality. Check on logging out; the user is logging out completely.

· Forgot password - Check if the forgot password option is available or not. And if it is there, does it work correctly. Also, check if it is prone to a security failure or URL manipulation

· Back and Forward buttons - Check how the application copes when using the browser's 'Back' and 'Forward' buttons.

· Remember me - Check if there is a "Remember me" option. And if it is present, then does it work as standard. Also, check what happens if the password is changed.

· Compatibility - Check the Login/Logout functionality with all possible valid/invalid cases in other browsers.

· Data - Check the username & password fields for data validation (Is there a minimum or maximum length of characters, boundary-values, what are the allowed characters, etc.).

· Error handling - Check how various errors are handled and displayed (for negative cases).

· JavaScript-off test - Check if the login form still works when JavaScript is disabled.

· 2FA Check - Check the login process when two-factor authentication is being provided; then test with valid/invalid token, test with valid/invalid backup code, test lockout procedure, and test recovery process.