**Experiment no.** 8

**Title**: Install of selenium tool and Demonstration of automated testing using selenium.

**Objective:** To learn automated testing using selenium.

**Theory :**

Selenium is a powerful open-source tool primarily used for automating web browsers. It provides a set of tools and libraries that enable testers and developers to automate interactions with web browsers for testing web applications. Selenium supports various programming languages such as Java, Python, C#, Ruby, JavaScript (Node.js), and more, making it accessible to a wide range of users.

### Components of Selenium:

1. **Selenium WebDriver:** WebDriver is the core component of Selenium that provides a programming interface for creating and executing test scripts. It allows you to control web browsers programmatically, interact with web elements, simulate user actions, and perform various testing tasks.
2. **Selenium IDE (Integrated Development Environment):** Selenium IDE is a browser extension used for recording and playback of user interactions with web applications. It provides a simple graphical interface for creating automated tests without writing code. While it's useful for quick test creation and prototyping, WebDriver is more powerful and flexible for complex testing scenarios.
3. **Selenium Grid:** Selenium Grid is a distributed testing framework that allows you to run tests in parallel across multiple browsers, operating systems, and machines. It enables efficient utilization of resources and reduces test execution time by running tests concurrently on different environments.

To Install and explore selenium for automated testing follow these steps:

1. Install Java Development Kit (JDK)
2. Download Selenium WebDriver

Navigate to the Selenium WebDriver official website (https://www.selenium.dev/downloads/) and download the WebDriver for your preferred programming language (e.g., Java).

Extract the WebDriver zip file to a location on your computer.

1. Set Up Development Environment

Choose an Integrated Development Environment (IDE) for writing Selenium tests. Popular choices include IntelliJ IDEA, Eclipse, and Visual Studio Code.

Download and install the IDE of your choice.

1. Configure Selenium WebDriver with Your IDE

If you're using Java, add the Selenium WebDriver jar files to your project's build path.

In Eclipse, create a new Java project and configure it to use the Selenium WebDriver jar files.

If you're using Maven or Gradle for dependency management, add Selenium WebDriver dependencies to your project's pom.xml or build.gradle file.

1. Set Up Browser Drivers

Selenium WebDriver requires browser drivers to communicate with web browsers. Download the browser driver for your preferred browser (e.g., ChromeDriver for Google Chrome, GeckoDriver for Firefox).

Place the browser driver executable in a location that is included in your system's PATH environment variable.

1. Verify Your Setup

Write a simple Selenium test script to open a web browser and navigate to a website.

Compile and execute the test script to ensure that Selenium WebDriver is working correctly.

1. Write simple script for to test user successful login to any web application.