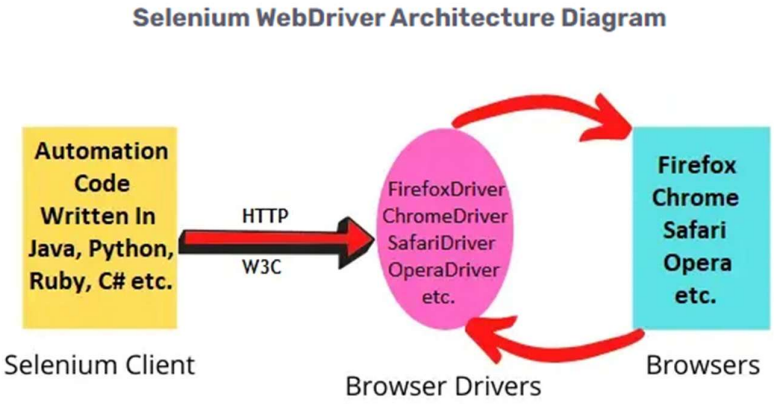
EXPERIMENT NO.: 10

Install and Explore Selenium for Automated Testing



Selenium Architecture

🎯 Aim:

Install and explore **Selenium** for automated web application testing using **Java and ChromeDriver**.

📖 What is Selenium?

Selenium is an open-source tool that allows **automated testing** of web applications across different browsers (Chrome, Firefox, etc.) and platforms.

You write scripts that open the browser, interact with it like a human, and test the website behavior.

✅ Tools Required:

Tool	Purpose	Where to Get it
JDK	Run Java programs	https://www.oracle.com/java/technologies/javase-downloads.html
Eclipse IDE	Java Development Environment	https://www.eclipse.org/downloads/

Tool	Purpose	Where to Get it
ChromeDriver	Connect Selenium to Chrome	https://sites.google.com/chromium.org/driver/
VS Code *(optional)*	Lightweight coding editor	https://code.visualstudio.com/

🔧 PART 1 — Eclipse (Recommended for Java + Selenium)

📁 Folder Structure (Maven Project)

```
SeleniumTest/
├── src/
│   └── main/
│       └── java/
│           └── DemoAppMain.java
├── pom.xml
```

🪜 Step-by-Step in Eclipse (Maven + Selenium)

✅ Step 1: Install Java JDK & Set Environment

1. Install JDK 17 or 21 from Oracle.
2. Set JAVA_HOME in System Environment Variables.
3. Verify in Command Prompt:

```
java -version
javac -version
```

✅ Step 2: Install Eclipse IDE (EE version is fine)

✅ **Step 3: Create Maven Project in Eclipse**

1. Open Eclipse → File → New → Other → Search Maven Project
2. Select template: maven-archetype-quickstart
3. Give project name: SeleniumTest

✅ **Step 4: Add Dependencies in pom.xml**

🔽 Replace your <dependencies> section with:

```
<dependencies>
 <!-- Selenium -->
 <dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-java</artifactId>
  <version>4.21.0</version>
 </dependency>

 <!-- JUnit -->
 <dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.13.2</version>
  <scope>test</scope>
 </dependency>
</dependencies>
```

Then right-click → Maven → Update Project.

✅ **Step 5: Download ChromeDriver**

1. Visit: https://sites.google.com/chromium.org/driver/
2. Match version with your installed Chrome.

3. Extract the downloaded file (e.g., chromedriver.exe) and place it in:

C:\SeleniumDrivers\chromedriver.exe

✅ **Step 6: Write Code – DemoAppMain.java**

📝 Go to src/main/java → Right click → New → Class → Name: DemoAppMain

Paste this short and clean code:

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

import org.openqa.selenium.chrome.ChromeDriver;


public class DemoAppMain {

  public static void main(String[] args) {

    System.setProperty("webdriver.chrome.driver",
"C:\\SeleniumDrivers\\chromedriver.exe");


    WebDriver driver = new ChromeDriver();

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


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


    driver.quit();

  }

}
```

▶️ **Step 7: Run the Program**

Right-click on file → Run As → Java Application

🟢 Output:

- Chrome browser opens → Goes to Google

- Terminal shows:
  Title is: Google

💡 Can We Do This in VS Code?

Yes! But VS Code does not have built-in Maven project wizards, so setup is slightly manual.

🪜 Steps in VS Code:

1. Install Extensions:

- Java Extension Pack
- Maven for Java

2. Create a Maven Project:

mvn archetype:generate -DgroupId=com.test -DartifactId=SeleniumTest -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false

cd SeleniumTest

code .

3. Add Dependencies in pom.xml (same as above)

4. Place chromedriver.exe in C:\SeleniumDrivers

5. Write Java code inside src/main/java/com/test/App.java:

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

import org.openqa.selenium.chrome.ChromeDriver;


public class App {

   public static void main(String[] args) {

      System.setProperty("webdriver.chrome.driver",
"C:\\SeleniumDrivers\\chromedriver.exe");


      WebDriver driver = new ChromeDriver();

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

```
        System.out.println(driver.getTitle());
        driver.quit();
    }
}
```

6. Run via VS Code Terminal:

```
mvn compile
mvn exec:java -Dexec.mainClass="com.test.App"
```

🧠 Summary of Commands & Tools

Task	Tool/Command
Install JDK	Oracle Website
Check Java install	java -version, javac -version
Download ChromeDriver	chromedriver download
Create Maven project (VS Code)	mvn archetype:generate ...
Add Selenium & JUnit	In pom.xml dependencies
Run in Eclipse	Right-click → Run As → Java Application
Run in VS Code	mvn compile, mvn exec:java -Dexec.mainClass=...

🔚 Final Output

✓ Browser opens
✓ Navigates to Google
✓ Terminal prints:

```
Title is: Google
```
