Name: Kaushal Kotkar Department: Cyber Security

Div: BE-15 Roll No: 54

Subject: DSO

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Experiment No. – 4** | | | | |
| **Date of Performance:** | 05/08/2024 | | | |
| **Date of Submission:** | 12/08/2024 | | | |
| Program Execution/ formation/ correction/  ethical practices (06) | Timely Submission  (01) | Viva (03) | Experiment Total (10) | Sign with Date |
|  |  |  |  |  |

**Experiment No. 4**

**Aim:** To Setup and Run Selenium Tests in Jenkins Using Maven.

**Lab Outcome:** CSL701.2 Apply Jenkins to Build, Deploy and Test the Software Applications

**Theory:**

Selenium is a widely used tool for automating web applications, and Jenkins is a powerful automation server. Integrating Selenium tests with Jenkins using Maven streamlines the testing process.

**Selenium:** It's an open-source framework for automating web browsers, allowing testers to perform functional and regression testing on web applications.

**Jenkins:** An automation server that facilitates continuous integration and continuous delivery. It's used to automate building, testing, and deployment of software projects.

**Maven:** A build automation and project management tool. It simplifies project setup, handling dependencies, and building Java projects.

**Steps to Set Up and Run Selenium Tests in Jenkins Using Maven:**

**Step 1: Install Jenkins**

Download and install Jenkins on your server.

Set up and configure Jenkins according to your requirements.

**Step 2: Create a Jenkins Job**

Create a new Jenkins job (Freestyle project).

Configure the job to fetch your Selenium test project from a version control repository.

**Step 3: Install and Configure Maven**

Ensure Maven is installed on your Jenkins server.

Configure Maven settings and paths in Jenkins.

**Step 4: Build Selenium Tests Using Maven**

In the Jenkins job configuration, add a build step to execute Maven commands.

Use Maven commands like clean test to build and run Selenium tests.

**Step 5: View Test Results**

Configure the Jenkins job to generate test reports.

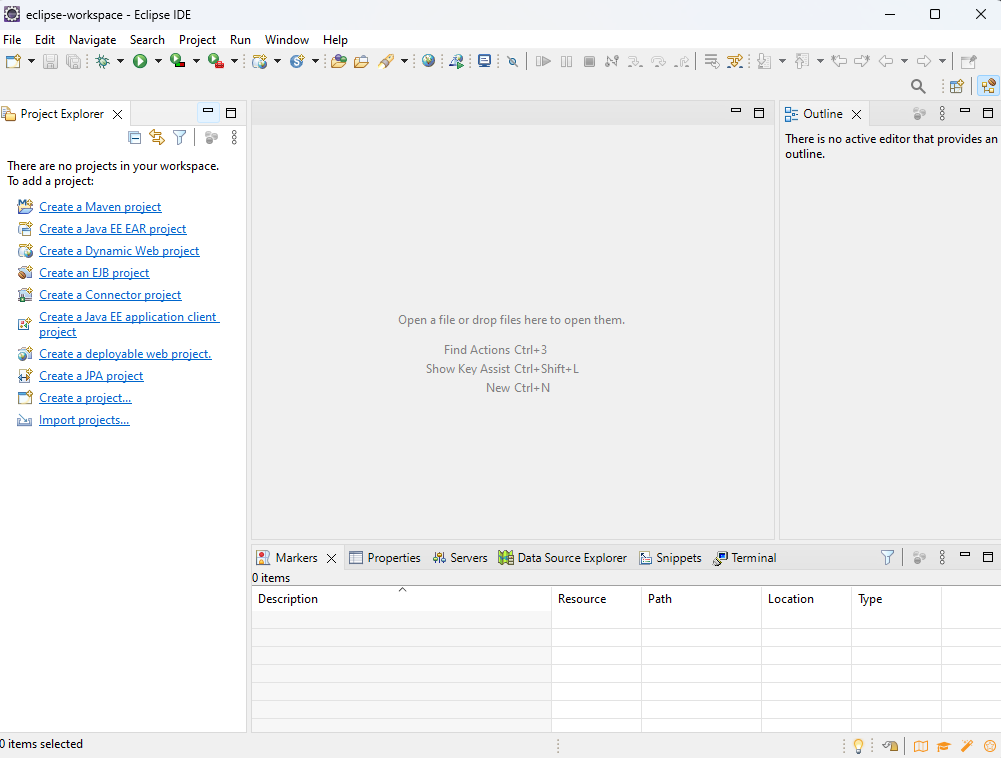
View test reports within Jenkins to analyze test results.

**Output:**

Installed Apache Maven.

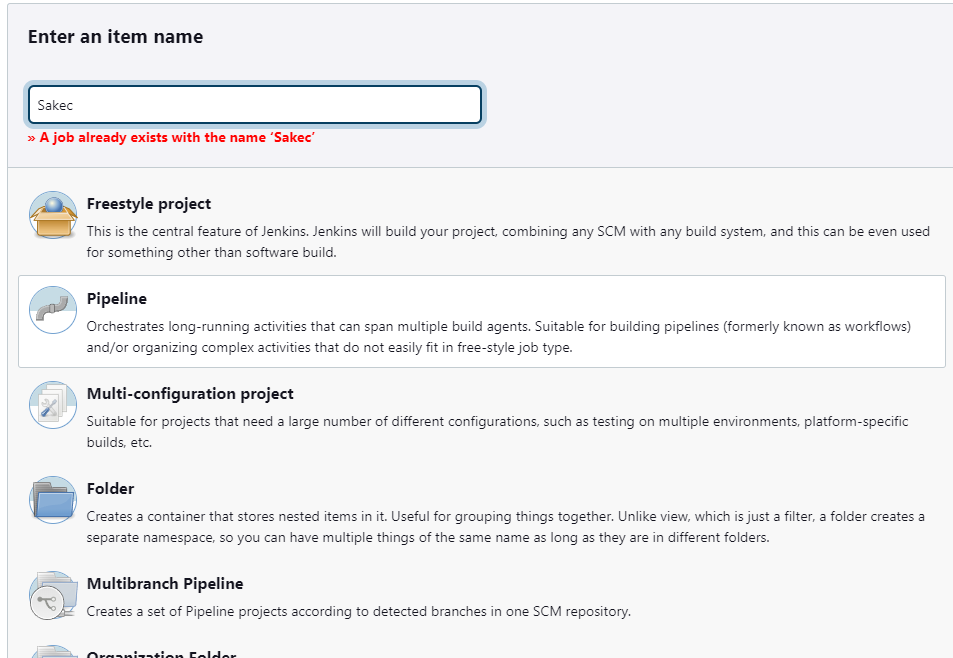
Installed Eclipse for Maven project development.

Added a Maven project in Eclipse.

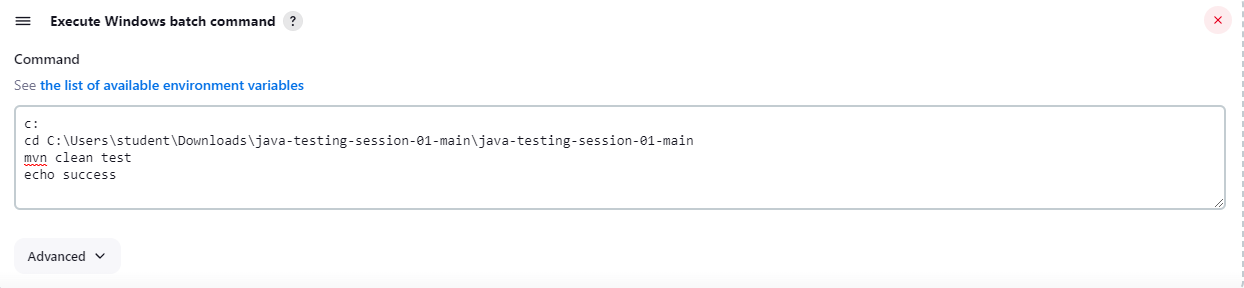


Created a Freestyle project in Jenkins.

Configured the Jenkins project with appropriate settings.



In the build steps, added the path to your Maven project and the command **mvn clean test**, and saved the configuration.

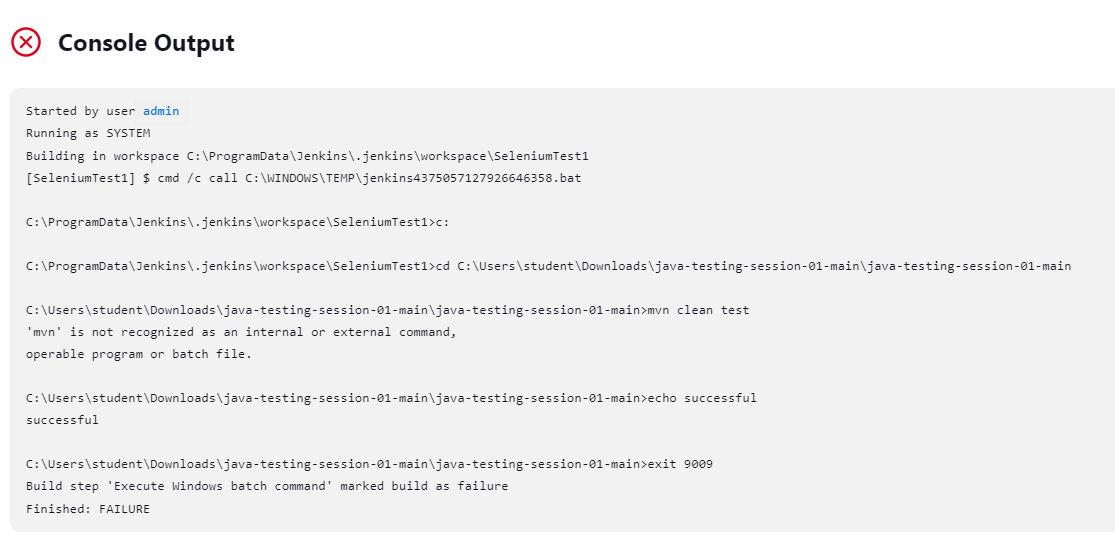




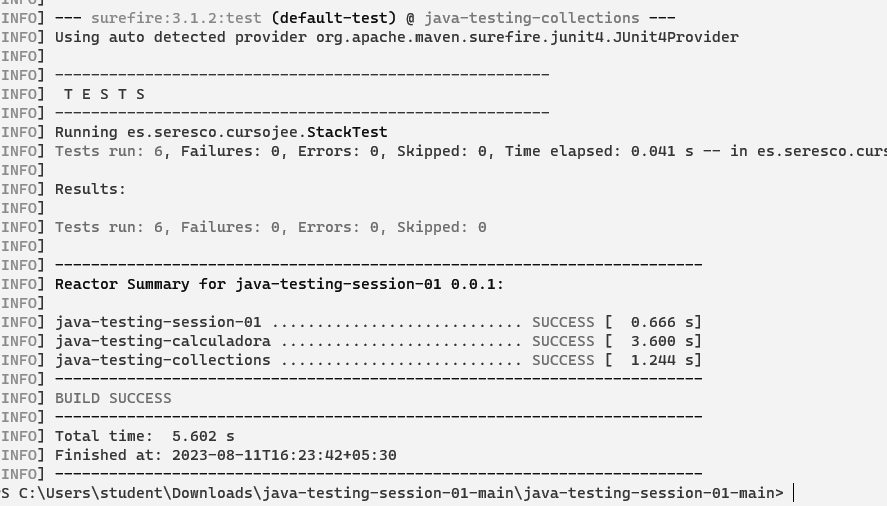
Used the "Build Now" option in Jenkins to initiate the build process.

Observed output for both success and failure cases.





Manually ran the command mvn clean test in the command prompt and received a successful result.



**Conclusion:**

Successfully configured Jenkins to seamlessly run Selenium tests using Maven, enhancing the testing process. This integration streamlines test automation and supports efficient continuous integration, enabling robust and reliable web application testing.