**SEPM**

**Experiment 7**

**Aim:**

To Setup and Run Selenium Tests in Jenkins Using Maven.

**Theory:**

Selenium is an open-source umbrella project for a range of tools and libraries aimed at supporting browser automation. It provides a playback tool for authoring functional tests across most modern web browsers, without the need to learn a test scripting language.

Selenium is one of the most renowned open-source test automation frameworks. It allows test automation of web apps or websites across different browsers & operating systems.

It also offers compatibility with multiple programming languages such as Java, JavaScript, Python, C#, and more, allowing testers to automate their website testing in any programming language that they are comfortable with.

Using the Selenium framework, testers are able to deliver test cycles faster by automating repeated test cases. When integrated with CI/CD pipeline, it can also help with a sturdy, bug-free release deployment pipeline.

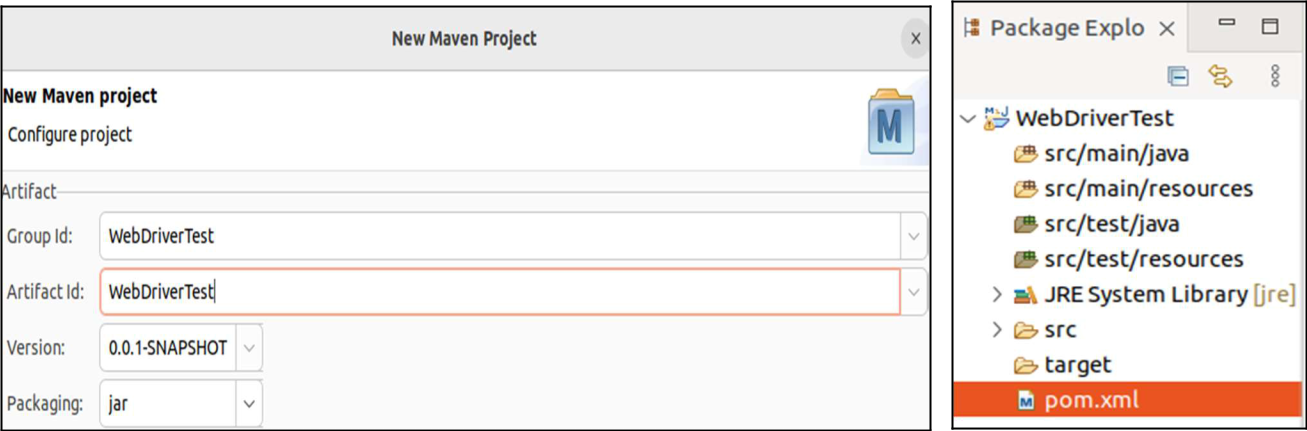
Selenium WebDriver allows you to interact directly with the browsers through your automation testing scripts. Java, PHP, C#, Python, Ruby, Perl, and Javascript are some programming languages it supports. The browsers Selenium WebDriver supports include Mozilla Firefox, Google Chrome version 12.0.712.0 and above, Internet Explorer, Safari, Opera version 11.5 and above, and HtmlUnit version 2.9 and above. As for operating systems, Selenium WebDriver supports Windows, Linux, Mac OS, and Solaris.

Selenium Grid has by far been the most useful component of the Selenium project. Selenium Grid allows parallel testing against various browsers & OS combinations through a Client-Server model. Here, the Server is known as the Hub which has multiple Clients to interact with. With Selenium Grid, you can connect a server to multiple remote machines which can then be used to run a browser automation script over multiple browsers + OS configurations, simultaneously.

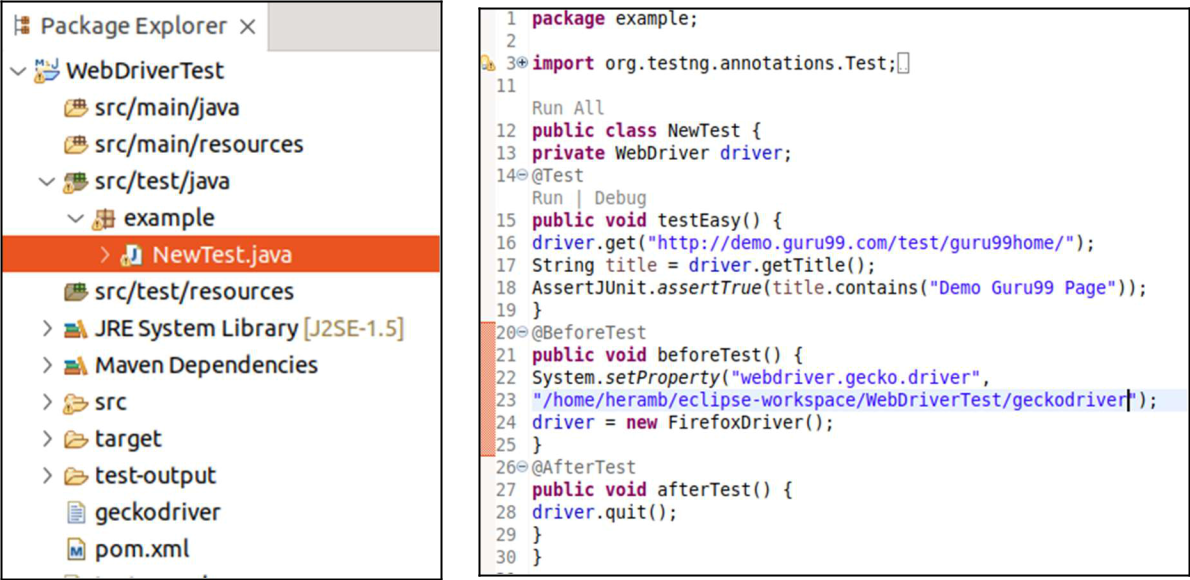
**Implementation:**

**Pre-requisites:** Eclipse With M2E & TestNg plugin, geckoDriver

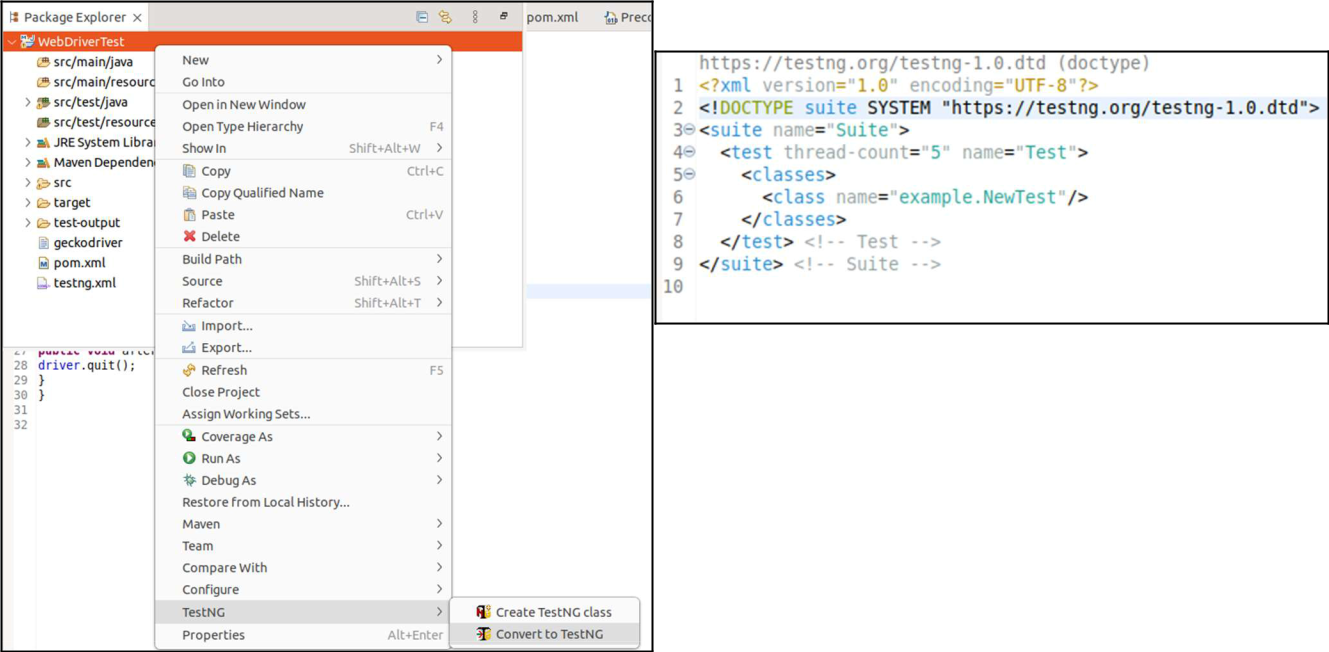
**Creating A Maven Project in Eclipse:**



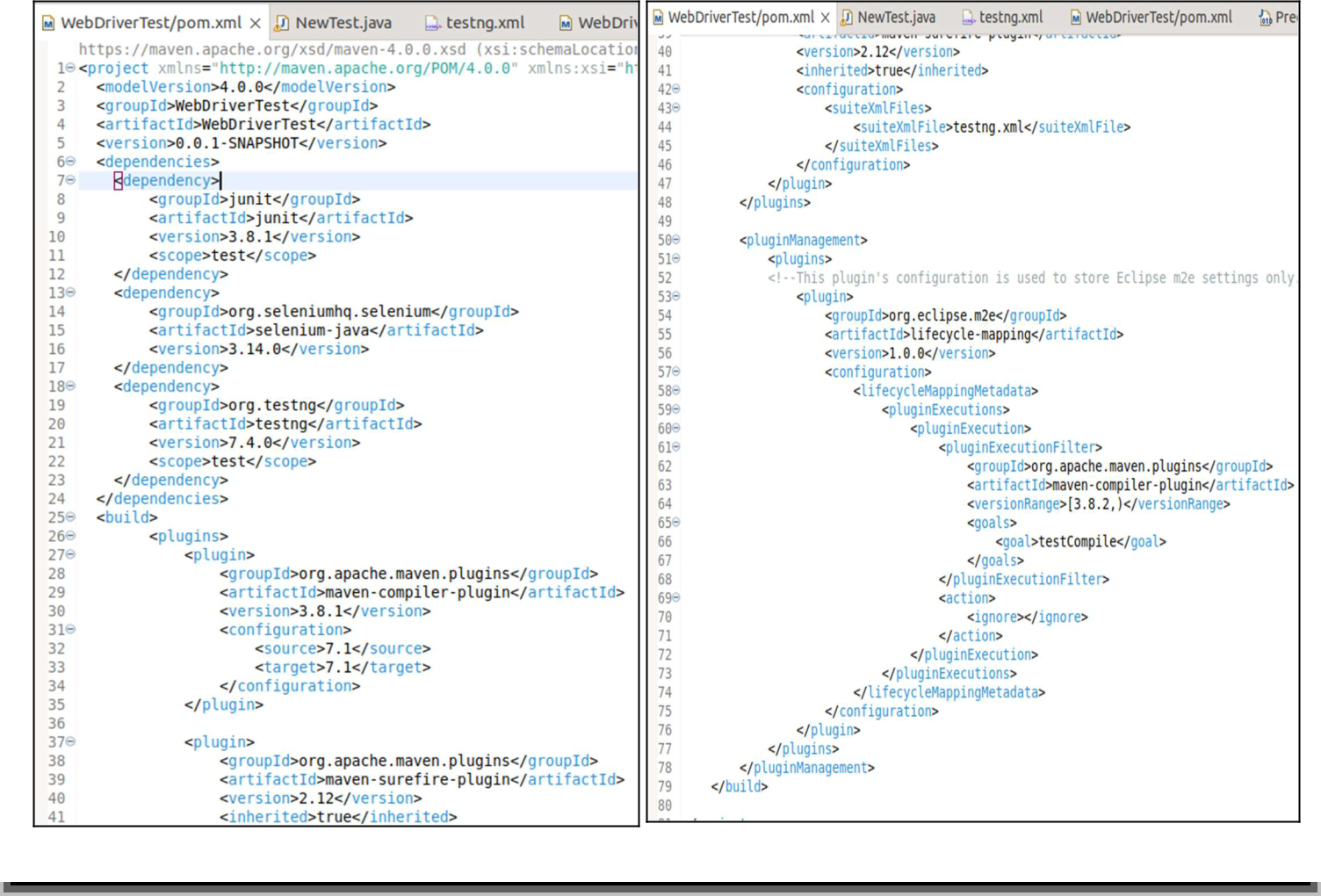
**Creating A Package And Creating A Java File In It:**

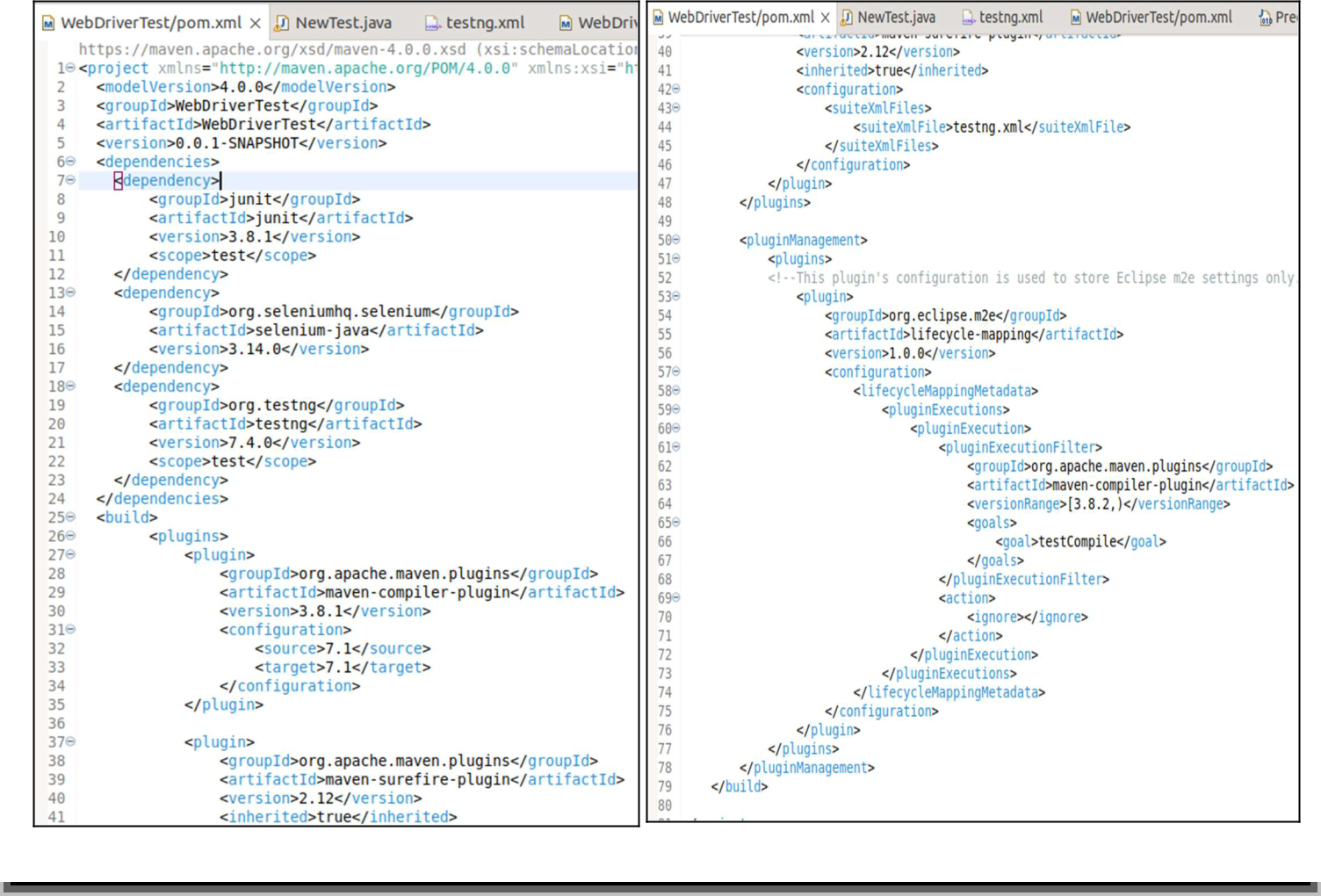


**Creating A TestNg.xml File:**

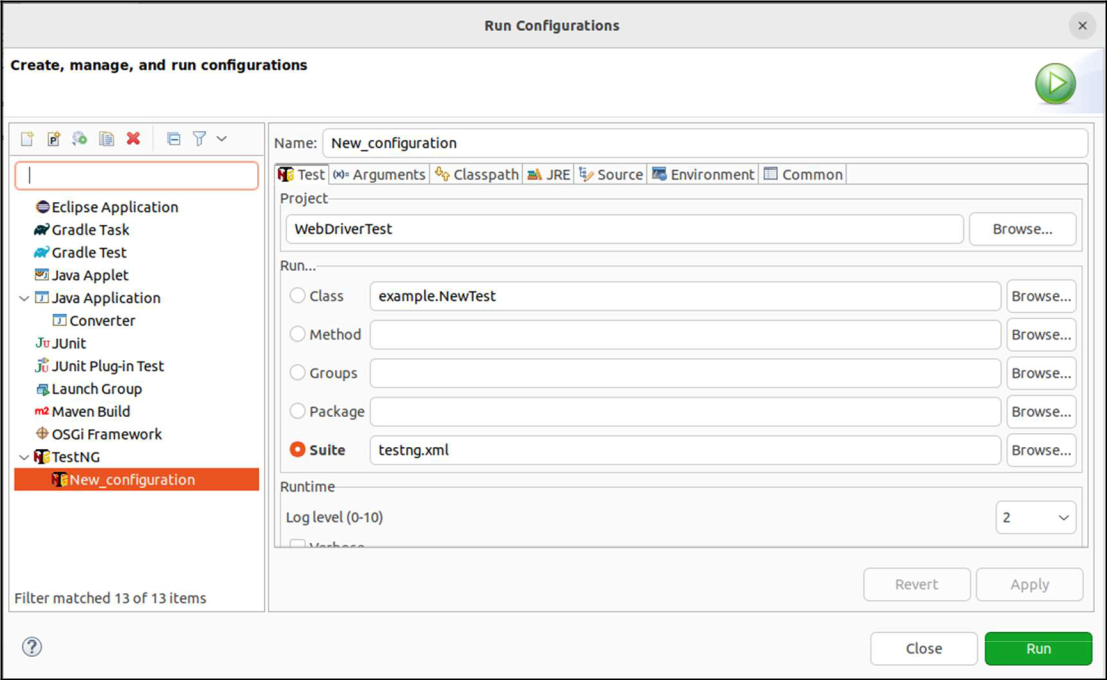


**Pom.Xml:**

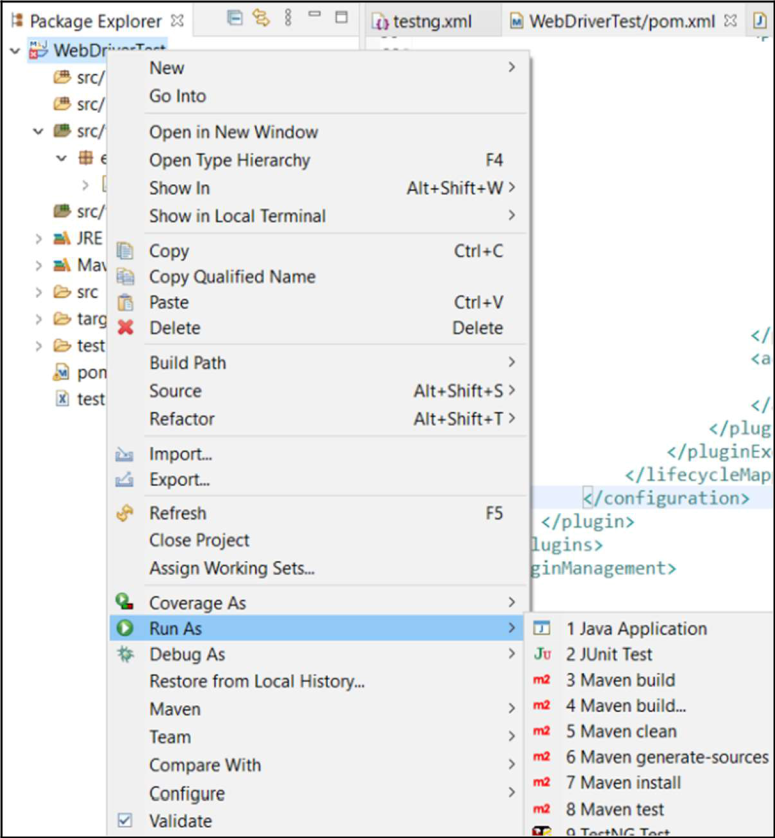




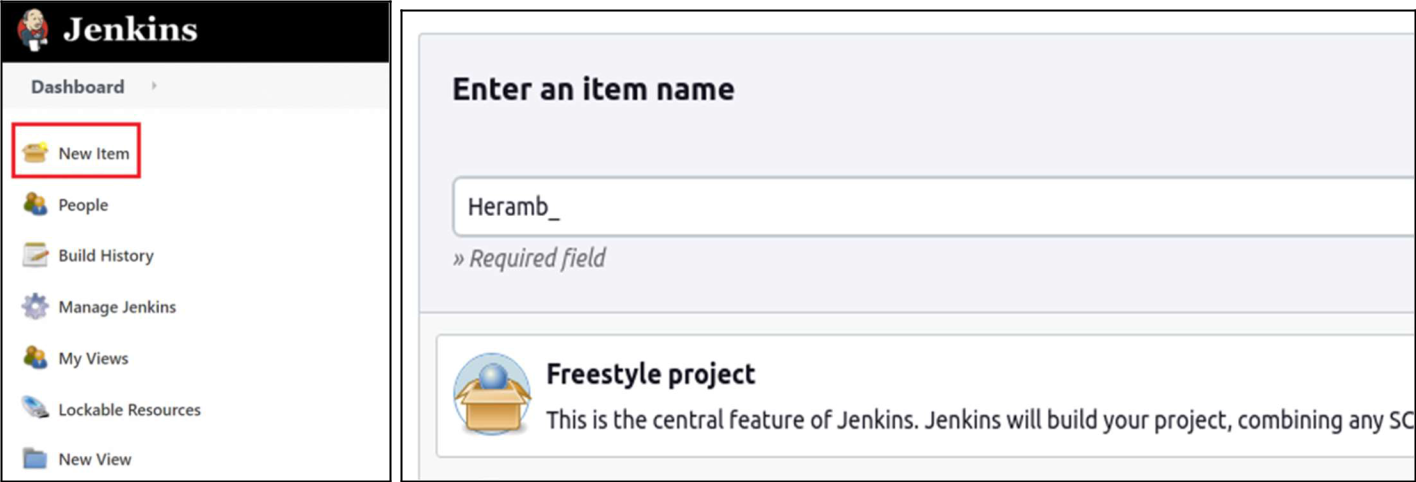
**Run>> Run Configuration>> Run**



**Running As Maven Test:**



**Setting Up Selenium on Jenkins:**

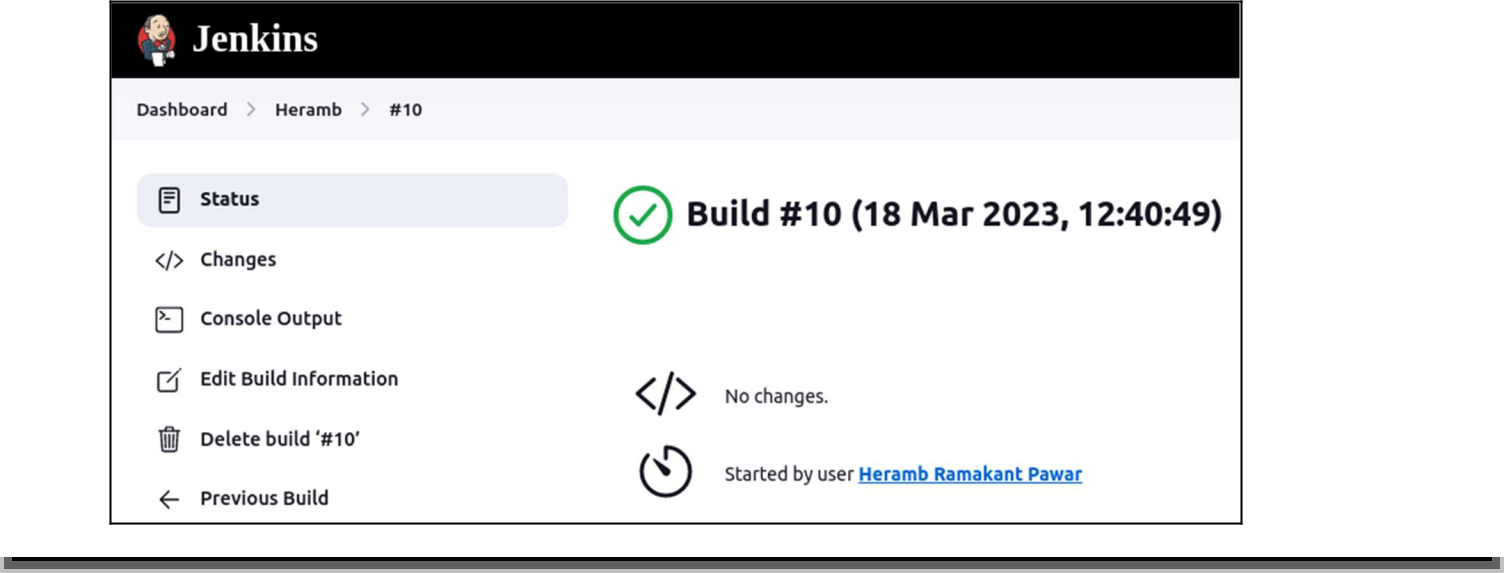


**Configure>> General >>Advance>>Use Custom Workspace**

**<add path to pom.xml in Directory option>**



**Build Now:**



**Conclusion:**

We have studied how to set up and Run Selenium Tests in Jenkins Using Maven. and its benefits. We also studied and installed Jenkins, along with its prerequisites and Maven.