Lesson 01 Demo 01

Create a JSP Page

Objective: To demonstrate creation of a JSP (Java Server Pages) page using Eclipse IDE

Tools Required: Eclipse IDE

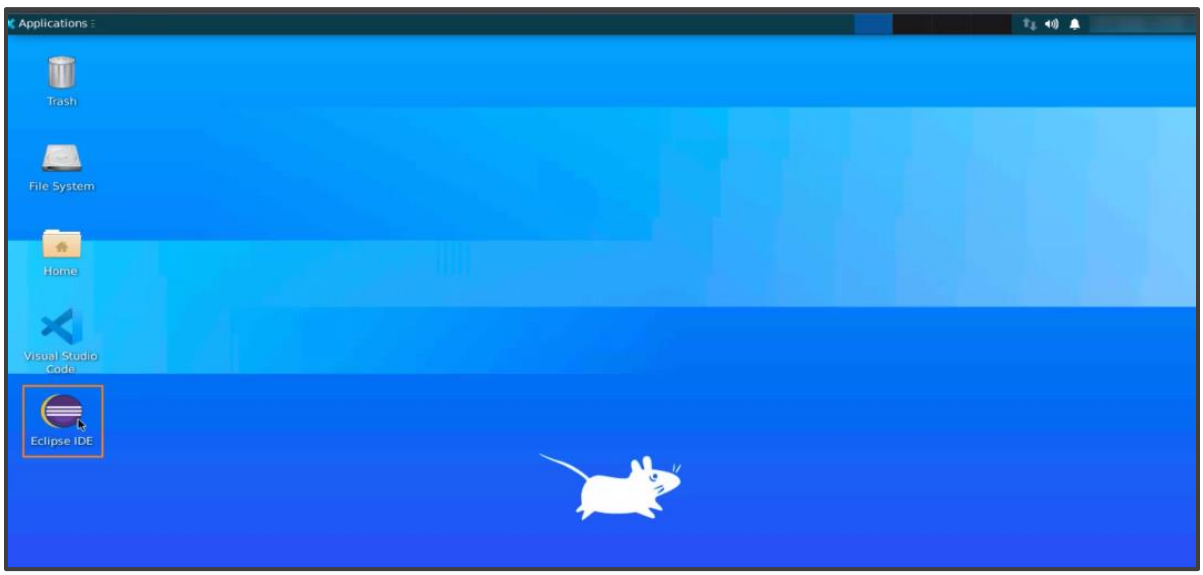
Prerequisites: None

Steps to be followed:

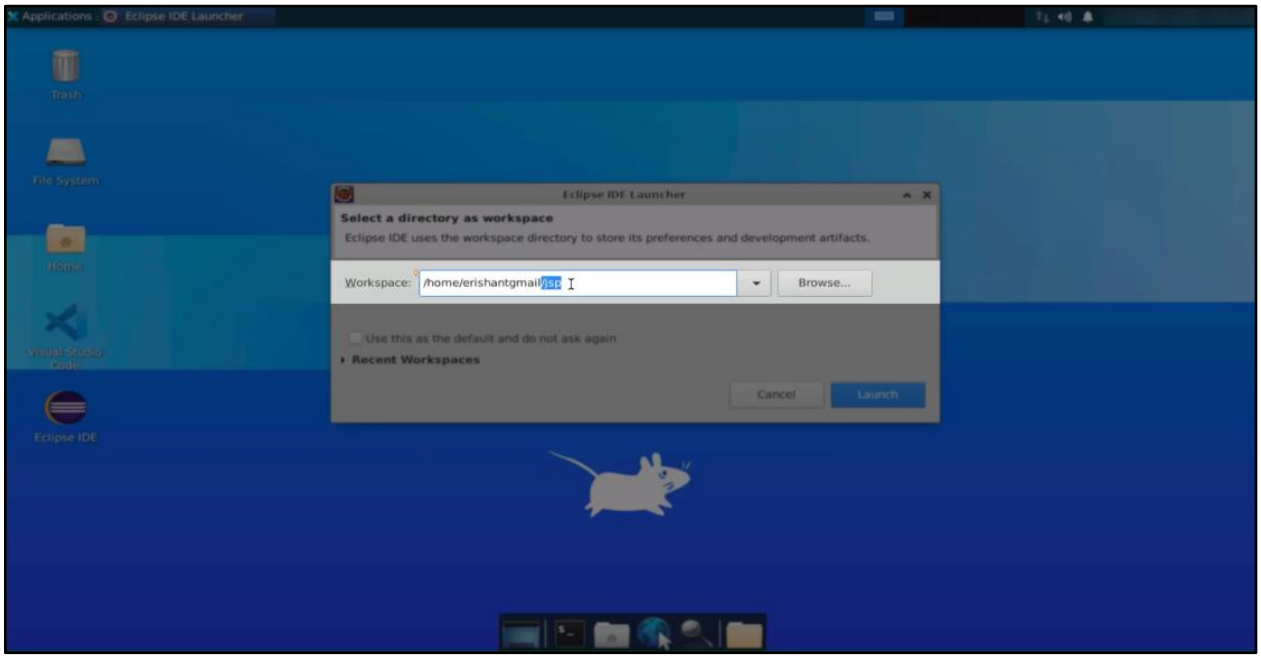
1. Create a new dynamic web project
2. Create a new JSP file
3. Incorporate changes in the previous operation
4. Create a new Maven project
5. Add the dependencies
6. Create a scriptlet tag

Step 1. Create a new dynamic web project

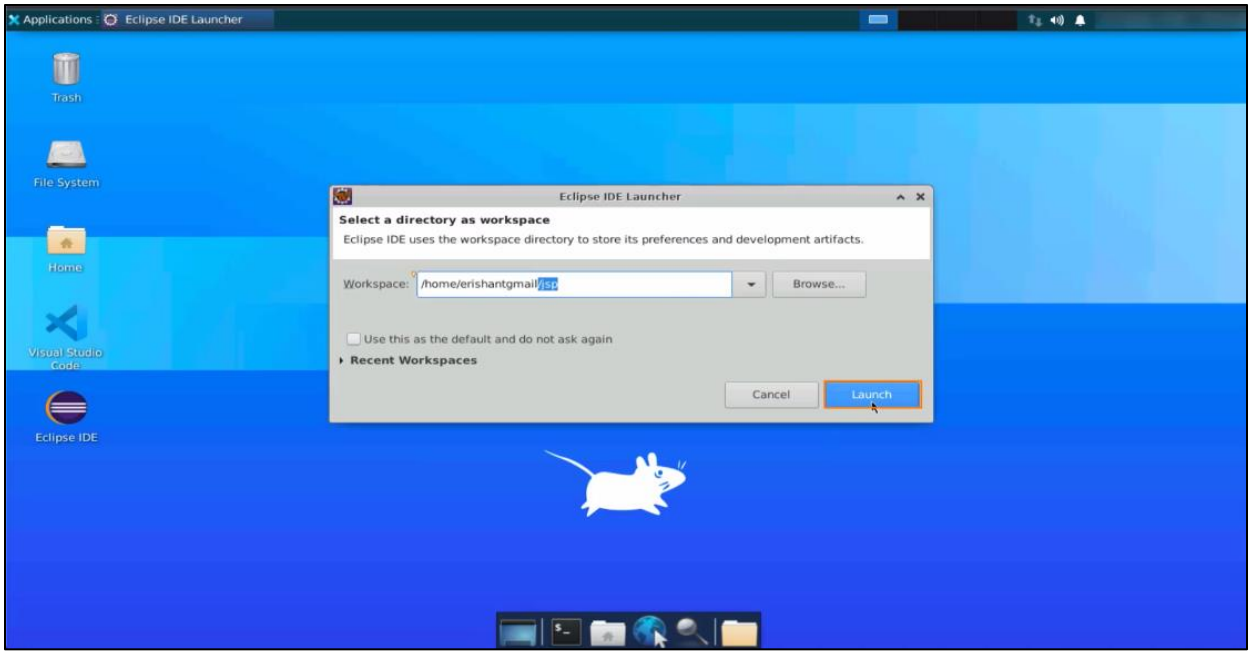
1.1 Open the **Eclipse IDE** in your lab



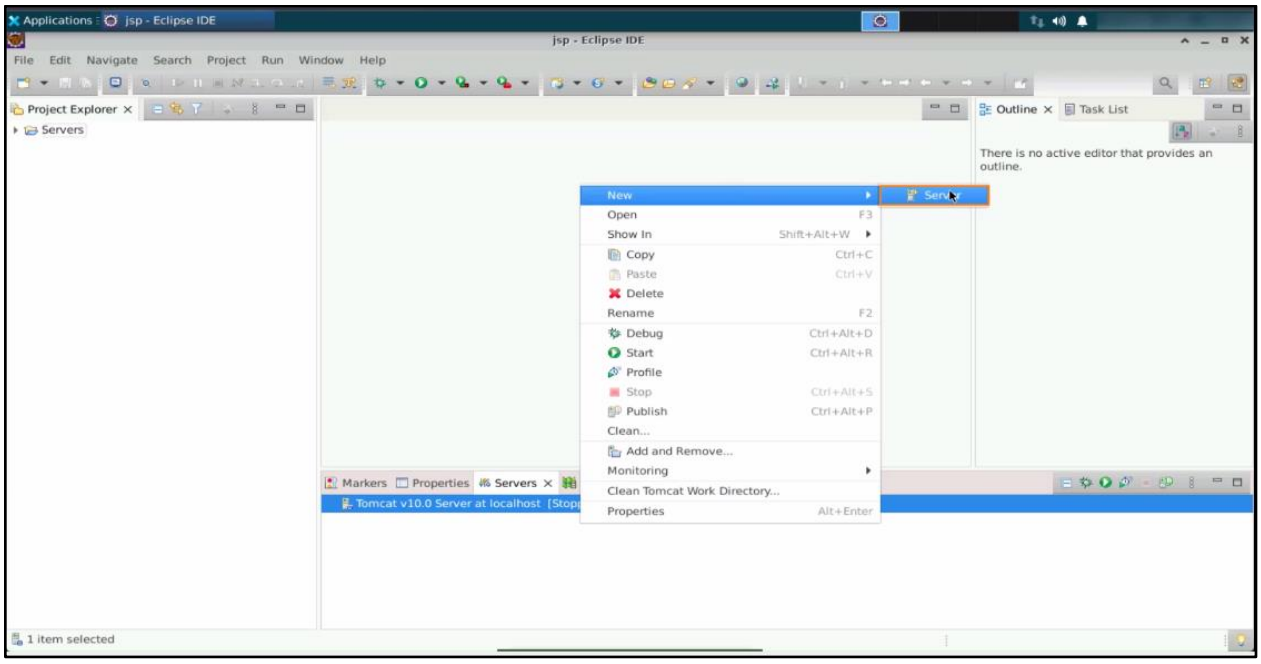
1.2 Set the workspace name as **jsp**



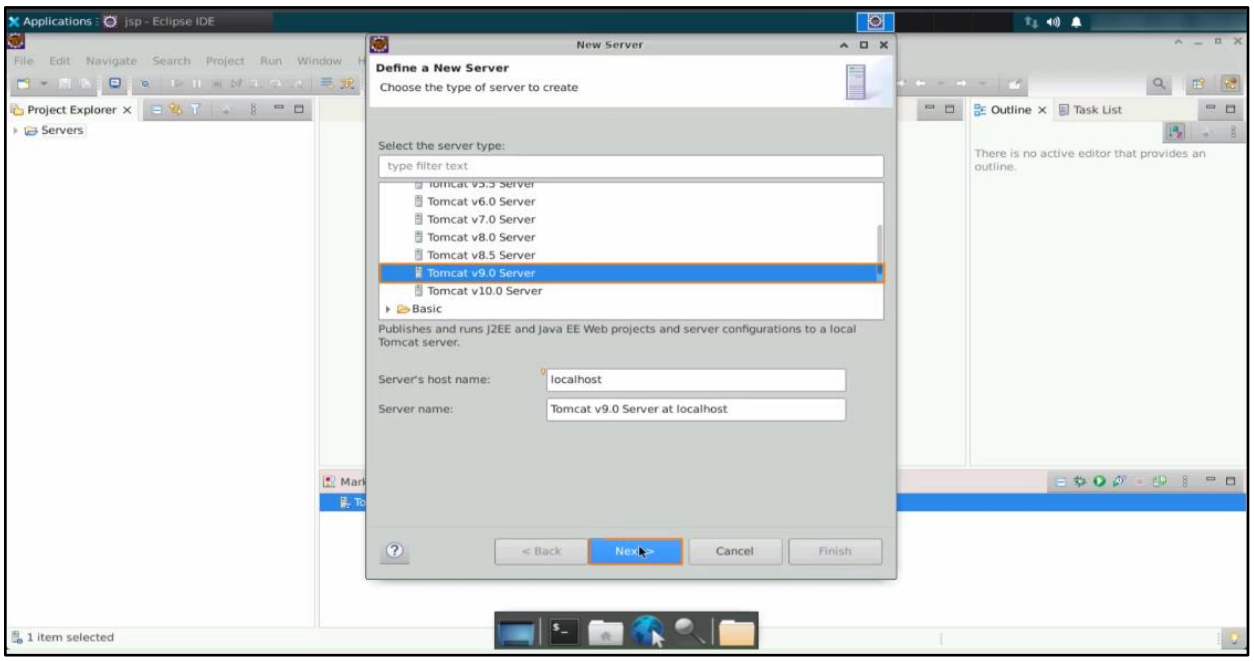
1.3 Click on **Launch** to open the IDE



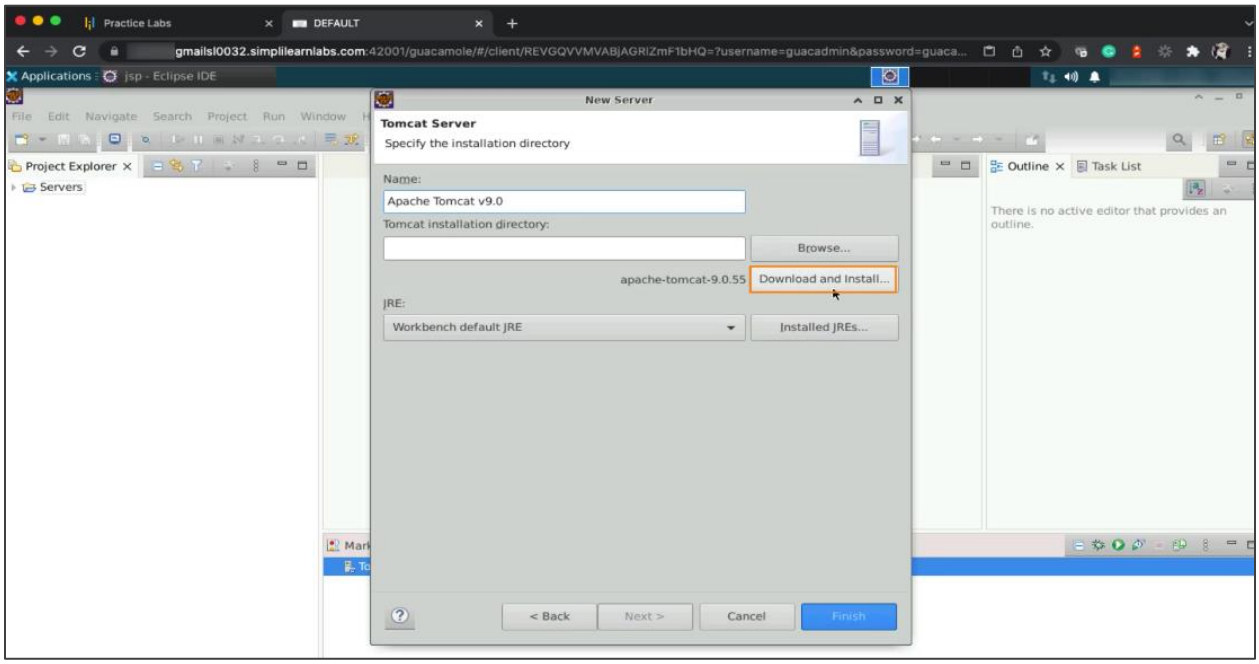
1.4 Right-click in the Project Explorer area and select **New**, then choose **Server** from the context menu



1.5 In the New Server dialog box, select **Tomcat v9.0 Server** and click **Next**

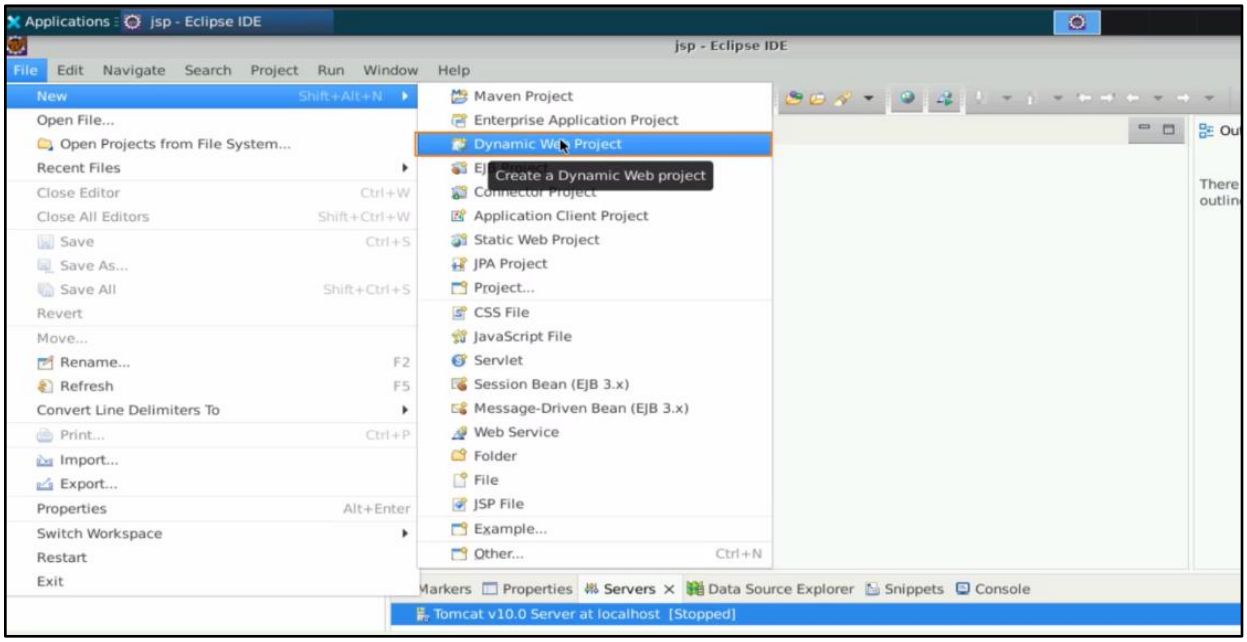


1.6 Choose the existing installation of Apache Tomcat or click on **Download and Install** to install it

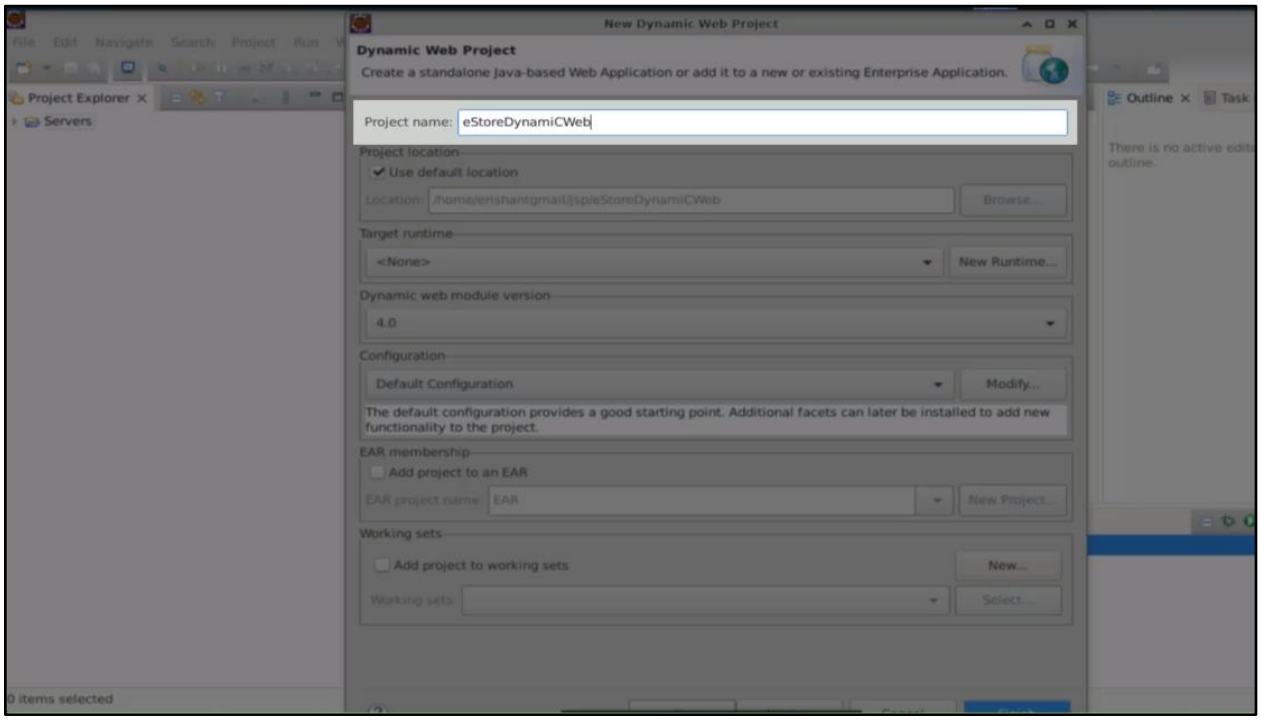


1.7 Create a new Dynamic Web Project by following the below path:

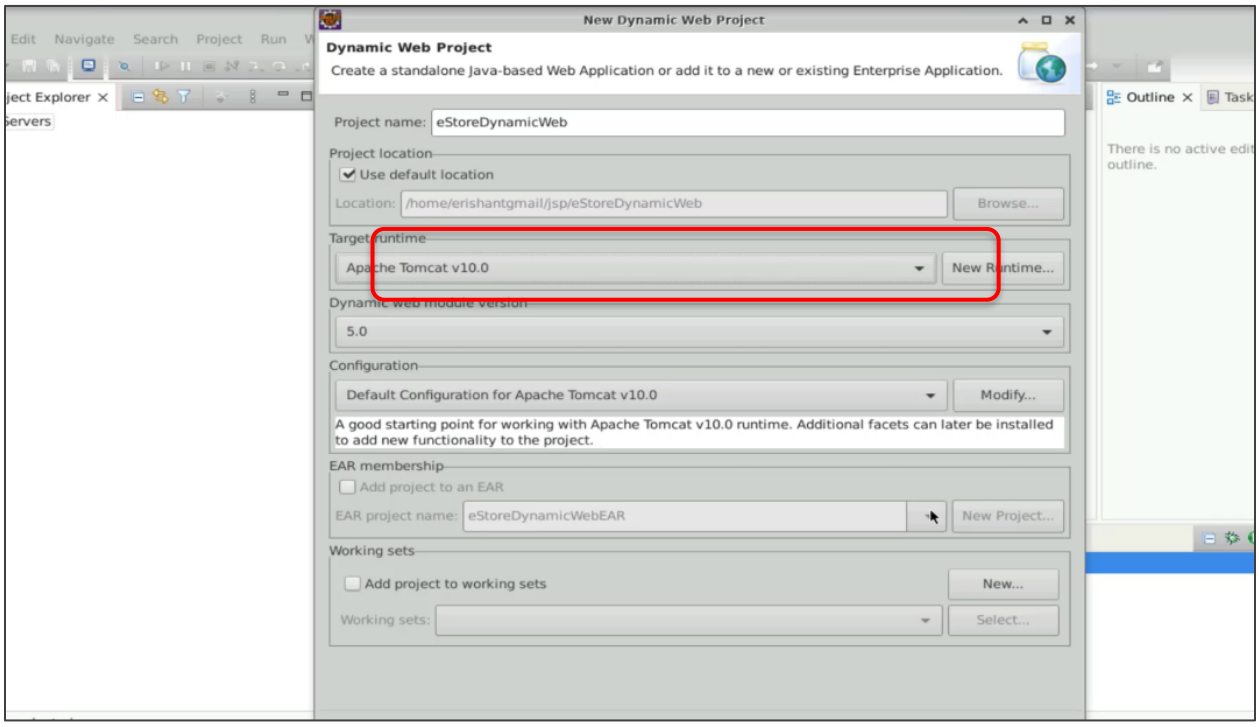
File->New->Dynamic Web Project



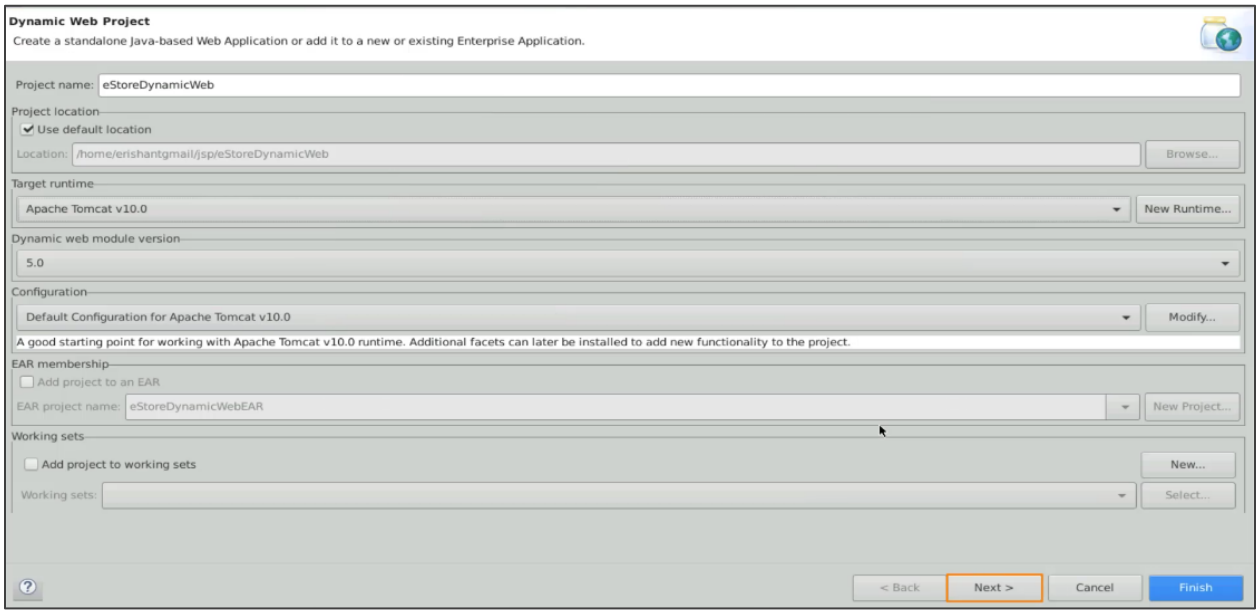
1.8 Name the project as **eStoreDynamicWeb**



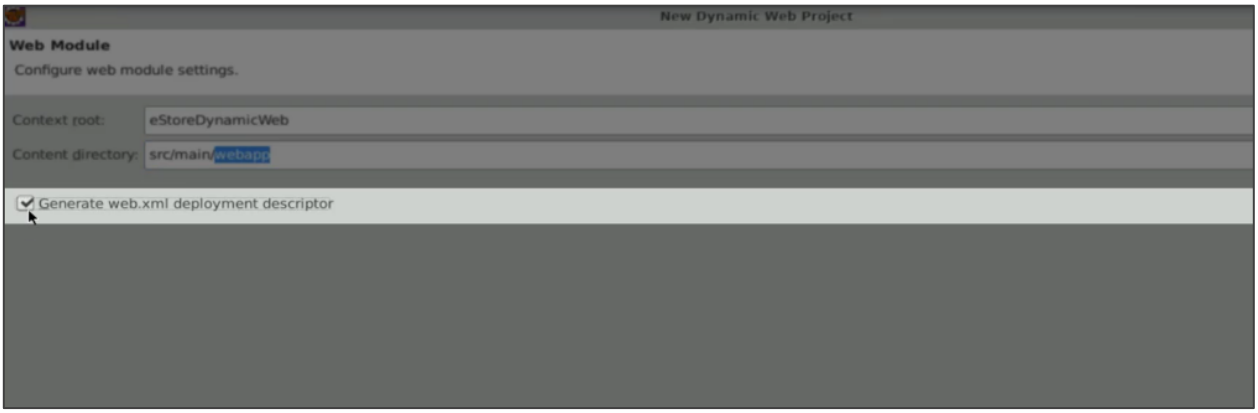
1.9 Keep the default configuration and set the Target runtime as **Apache Tomcat v10.0**



1.10 Click on **Next**

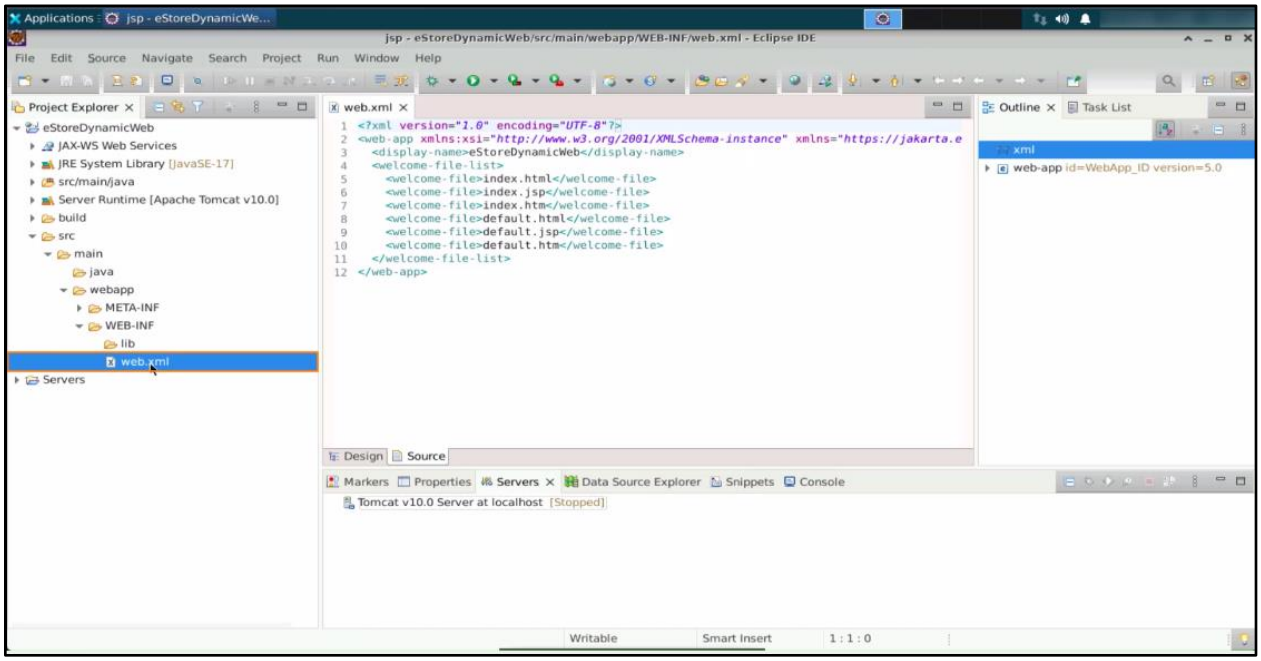


1.11 Check the option **Generate web.xml deployment descriptor** and click **Finish** to create the project

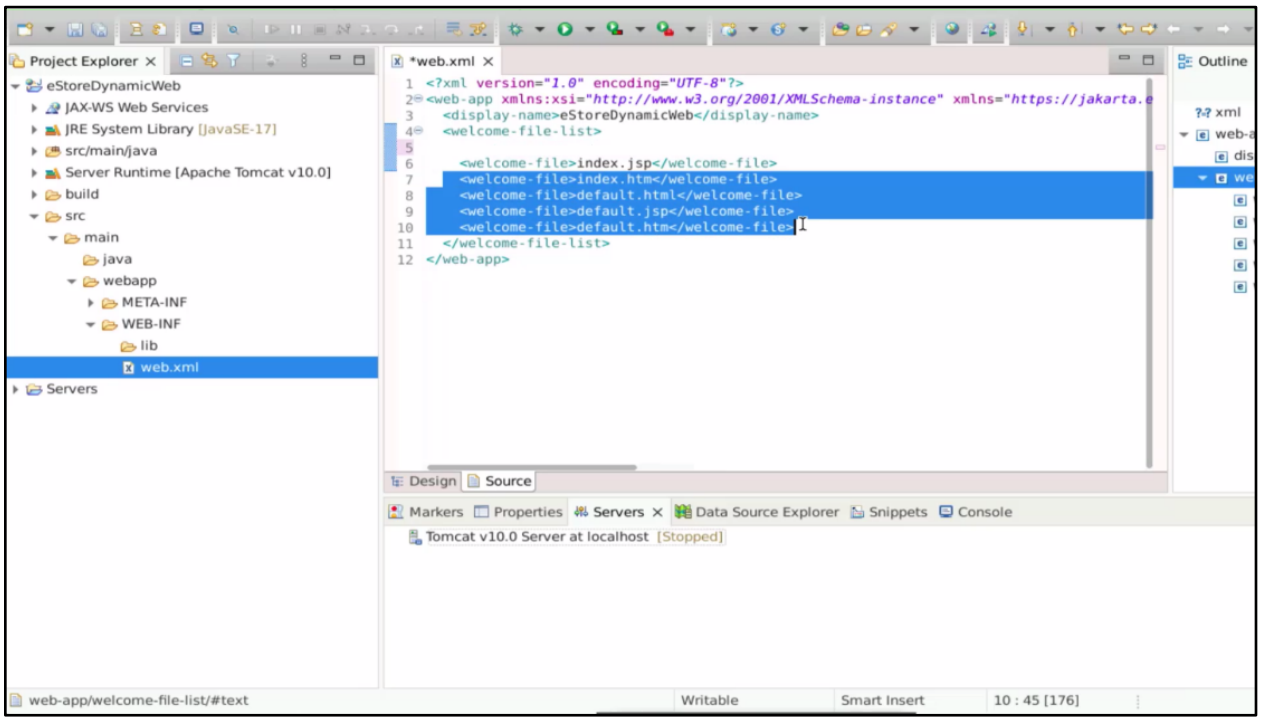


Step 2: Creating a new JSP file

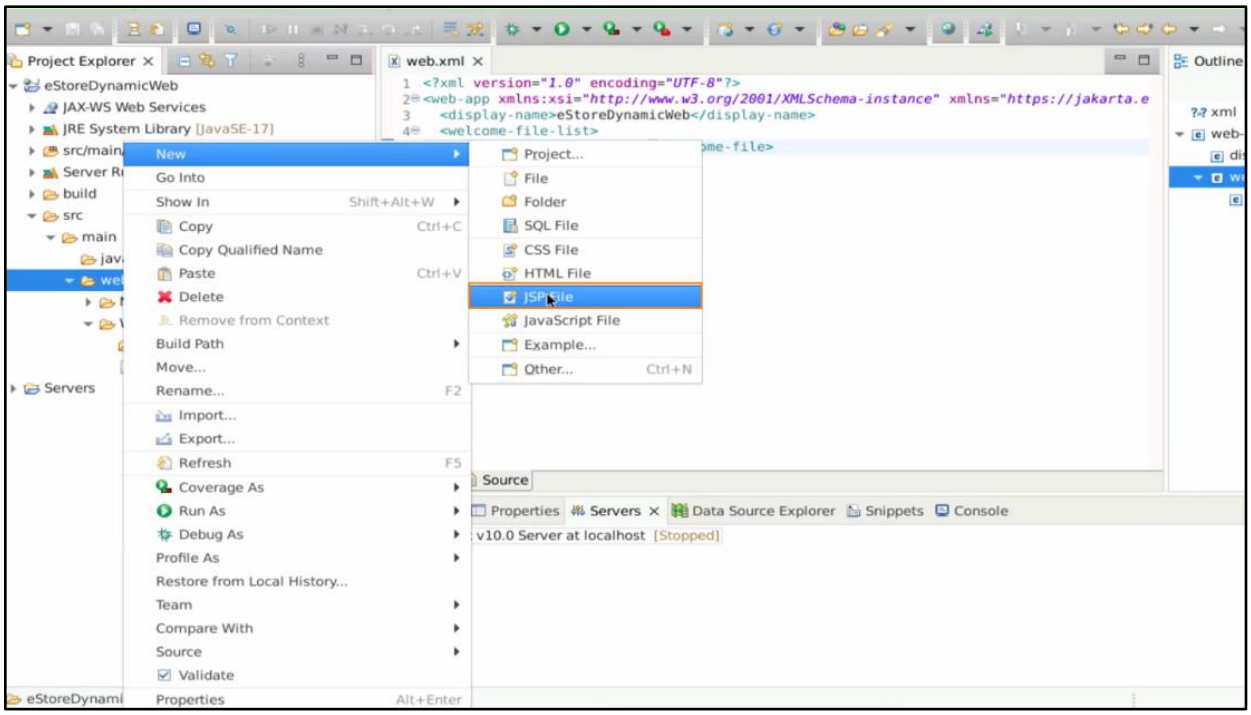
2.1 Open the **web.xml** file located in the **WEB-INF** directory of the project



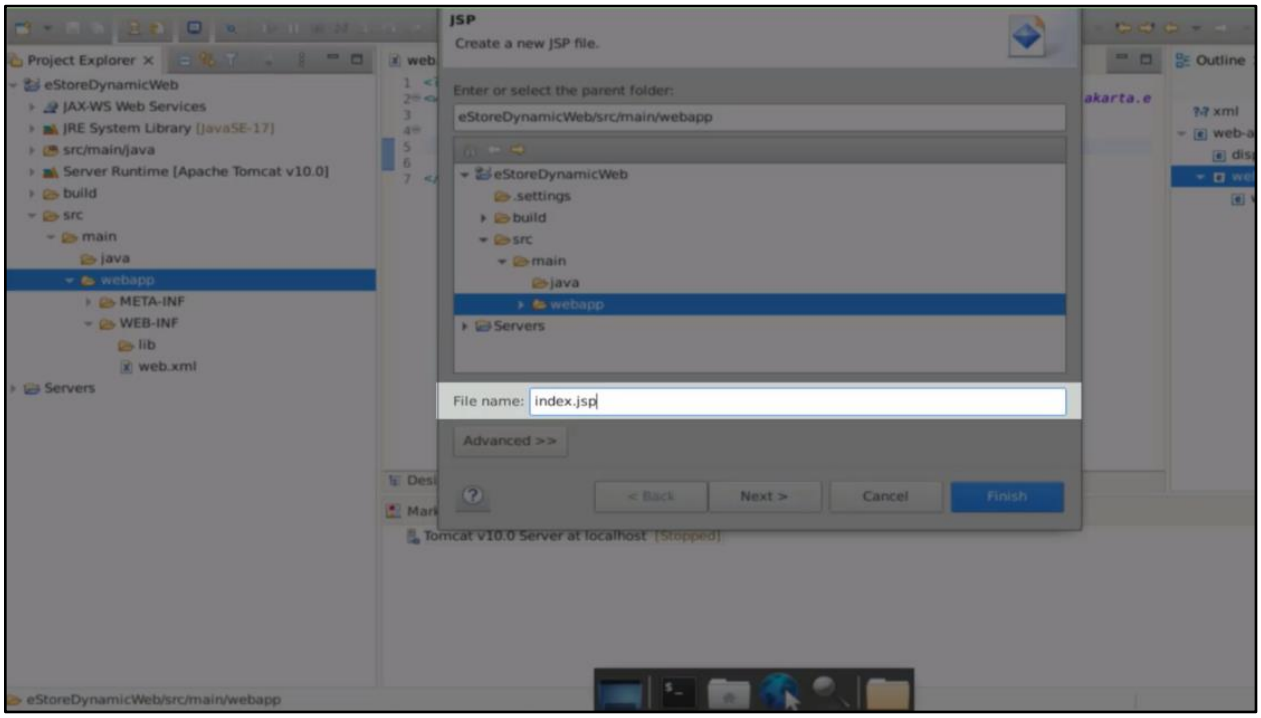
2.2 Remove all the files listed under the **welcome-file-list** section except the JSP file



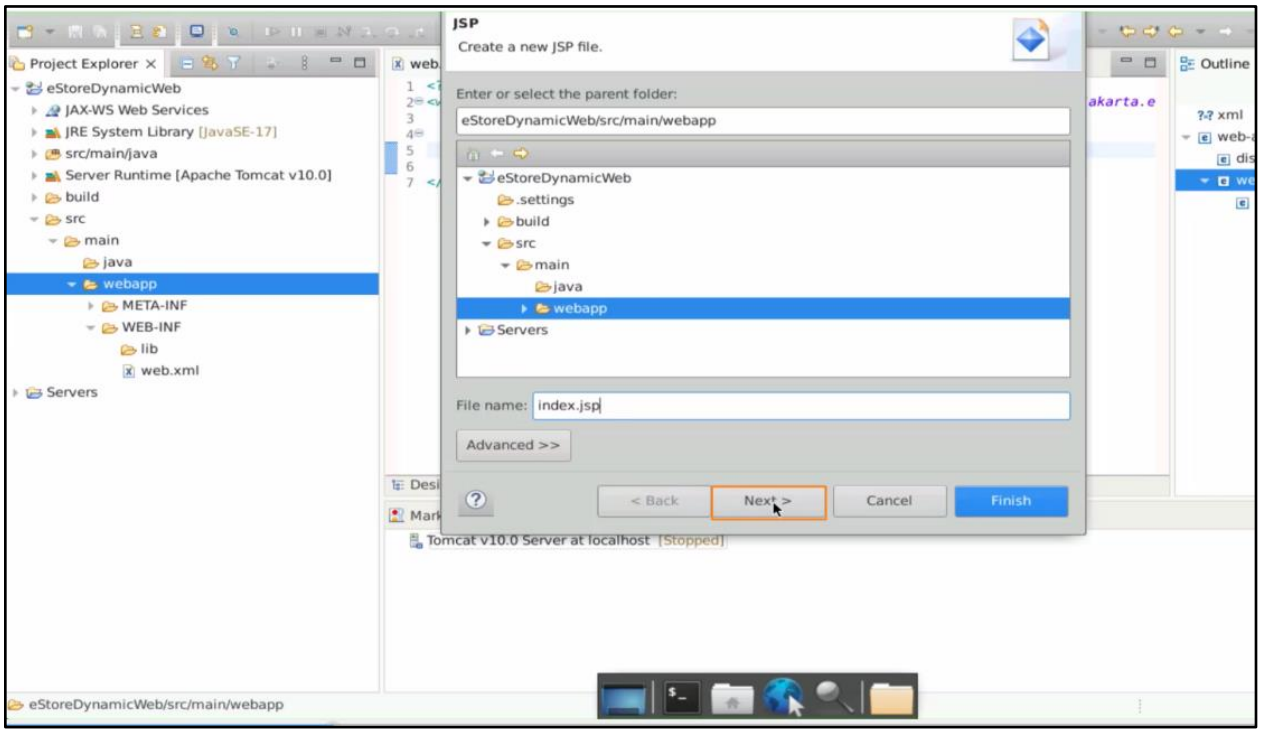
2.3 Right-click the webapp directory then select **New,** and choose **JSP File**

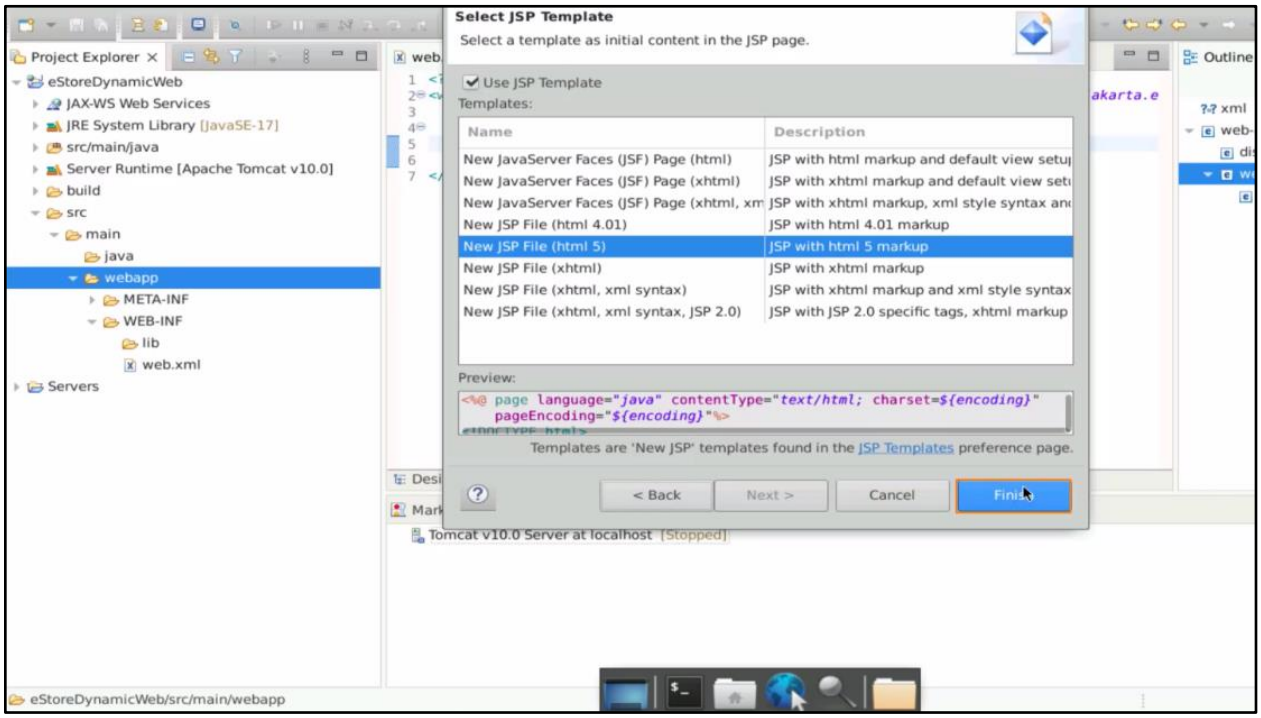


2.4 Name the file as **index.jsp**

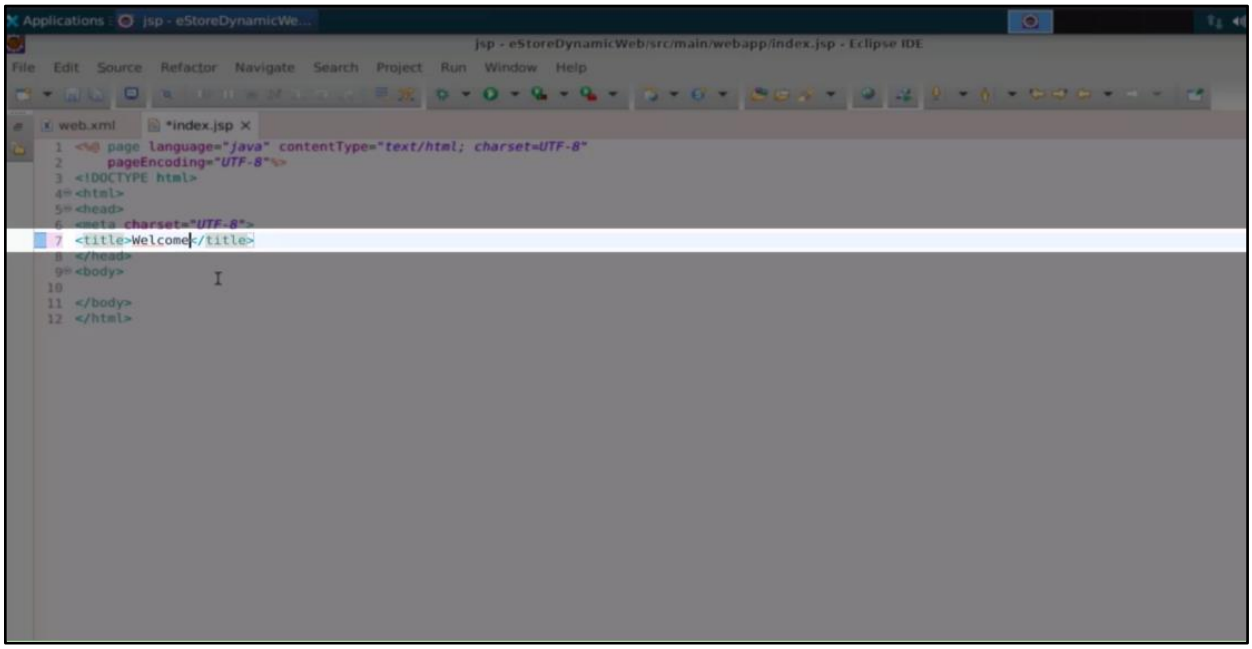


2.5 Click **Next** and then **Finish**



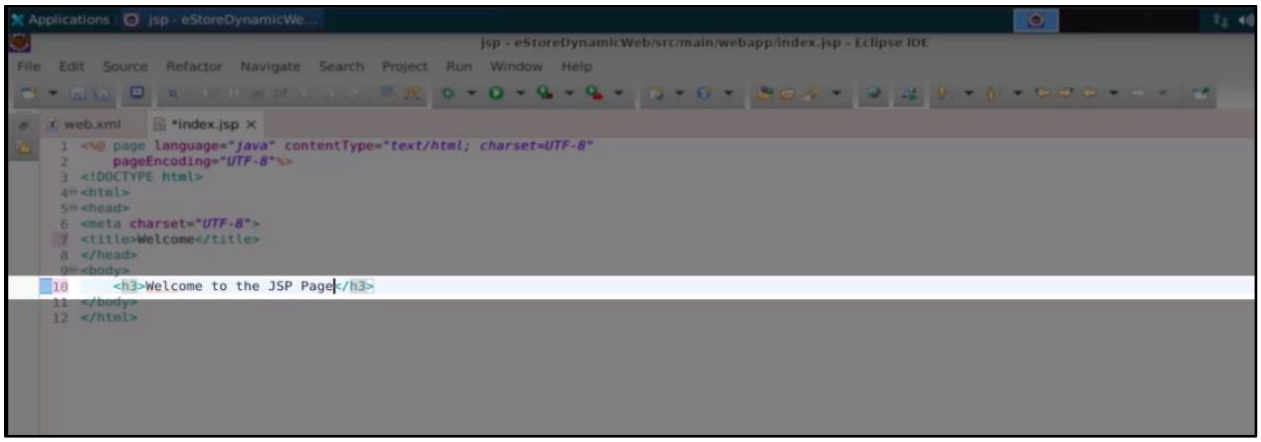


2.6 Open the **index.jsp** file and modify the title to **Welcome**

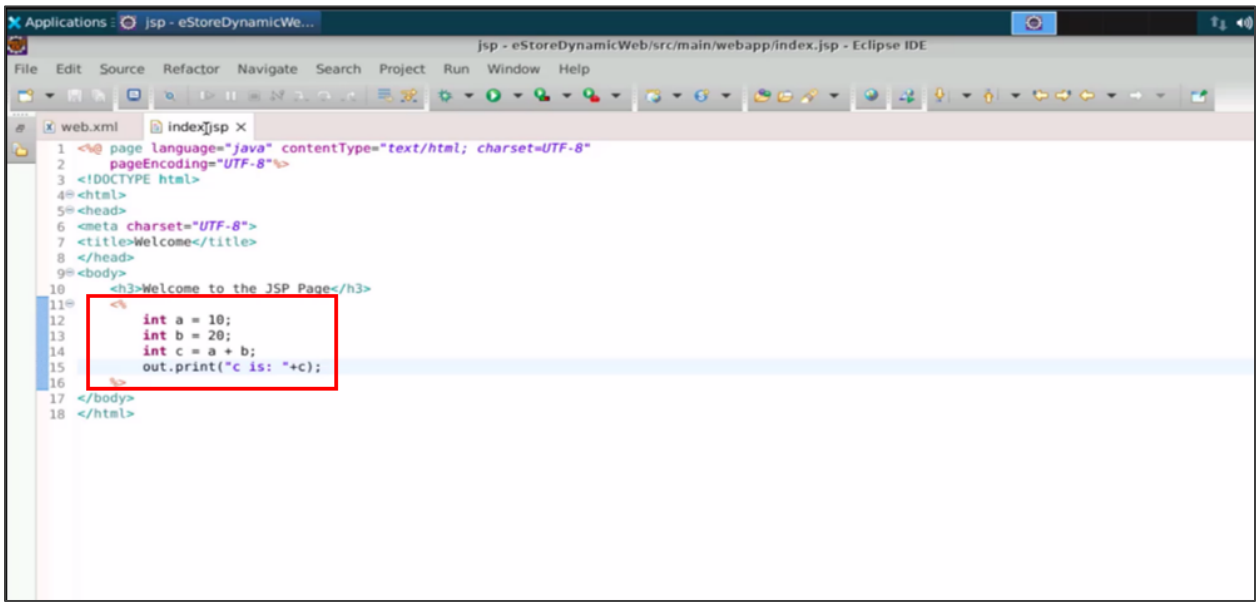


2.7 Add an h3 tag:

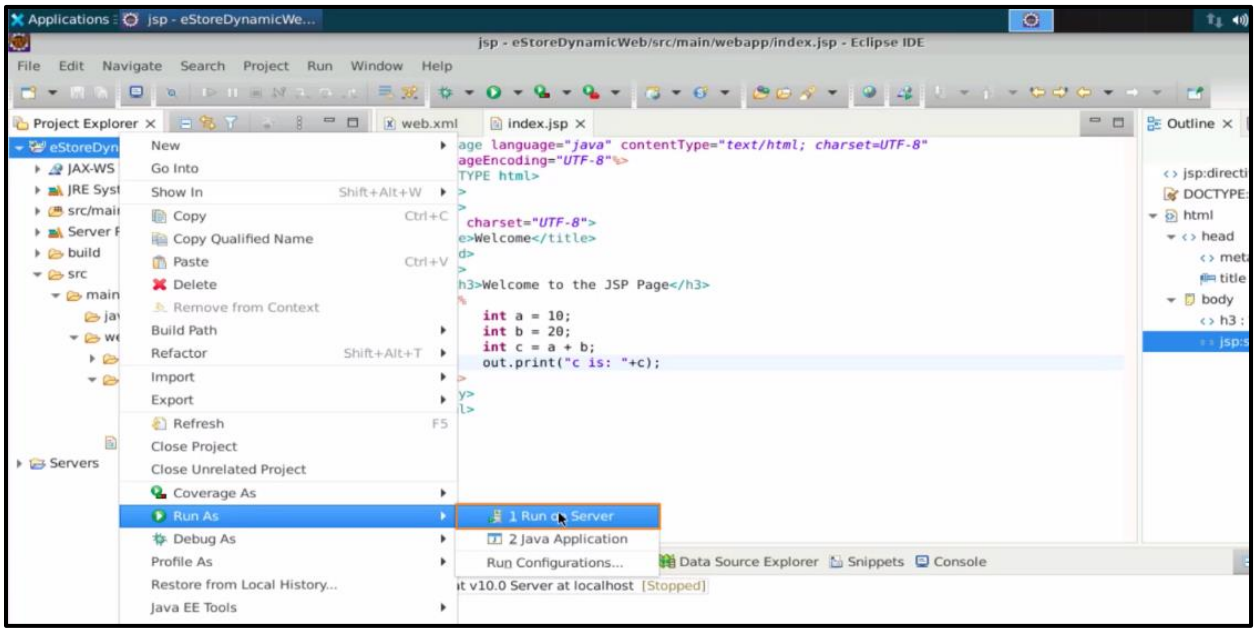
<h3>Welcome to the JSP Page</h3>



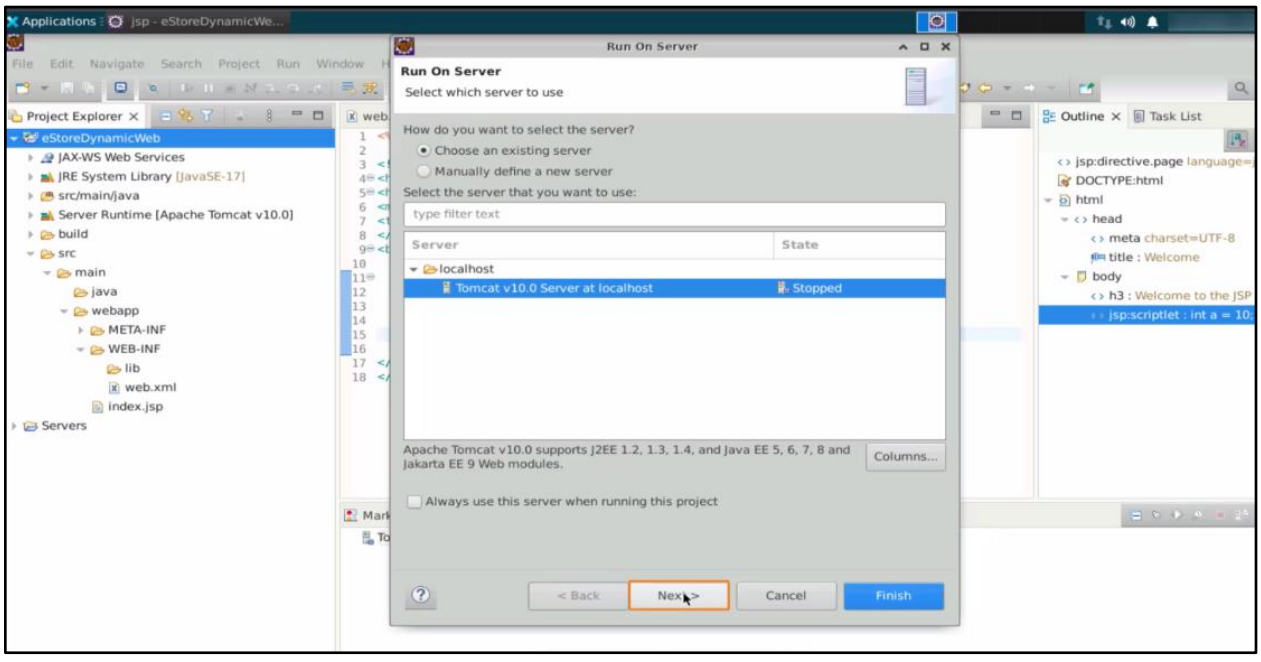
2.8 Create a scriptlet tag **<% %>** and add the desired logic or code within it



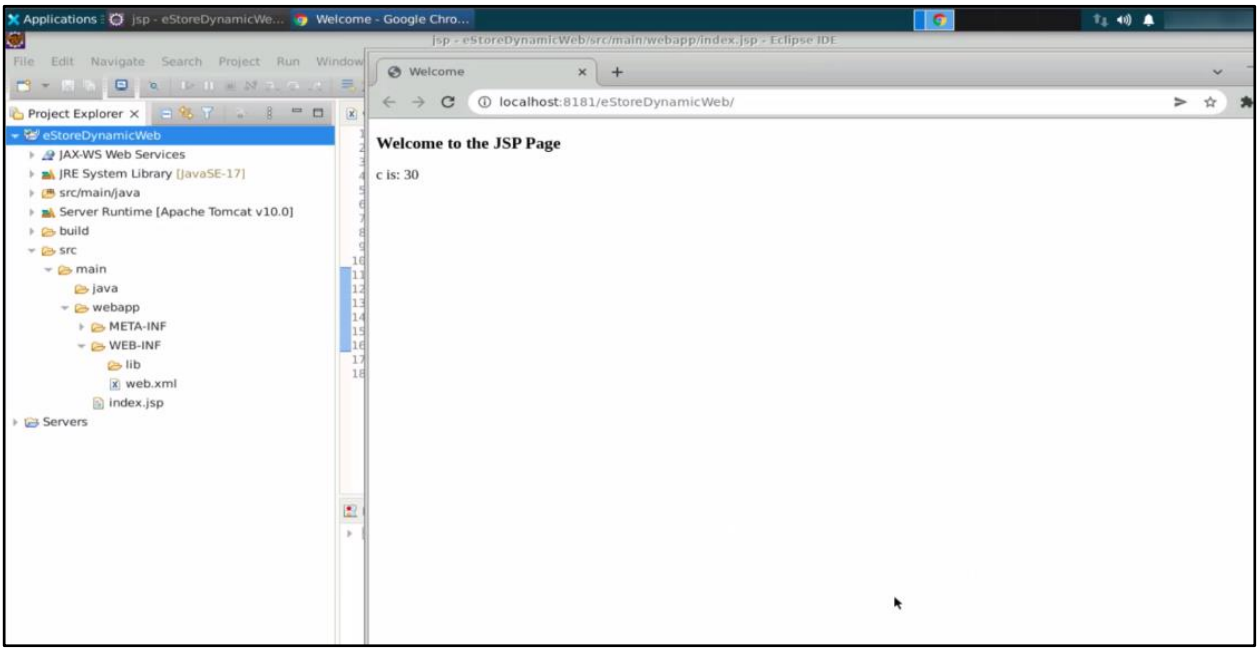
2.9 Right-click on the project, select **Run As**, and click on **Run on Server**



2.10 Choose the configured Tomcat server and click **Next**



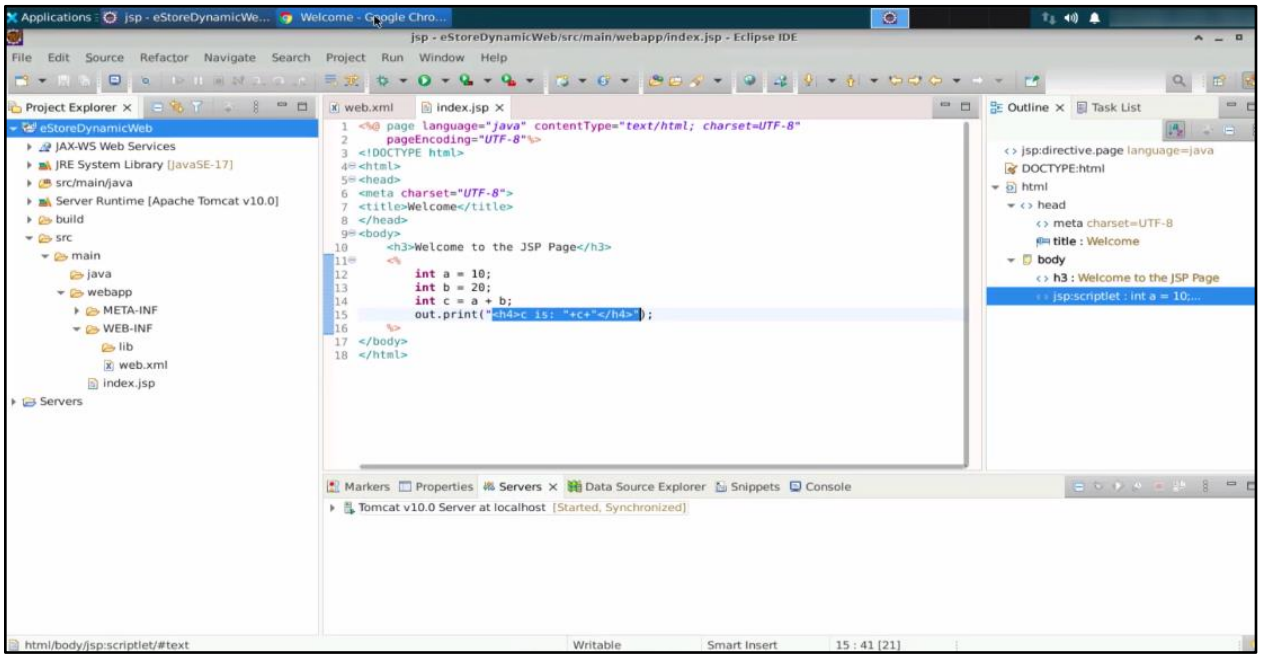
2.11 Review the server configuration and click **Finish** to run the project on the server



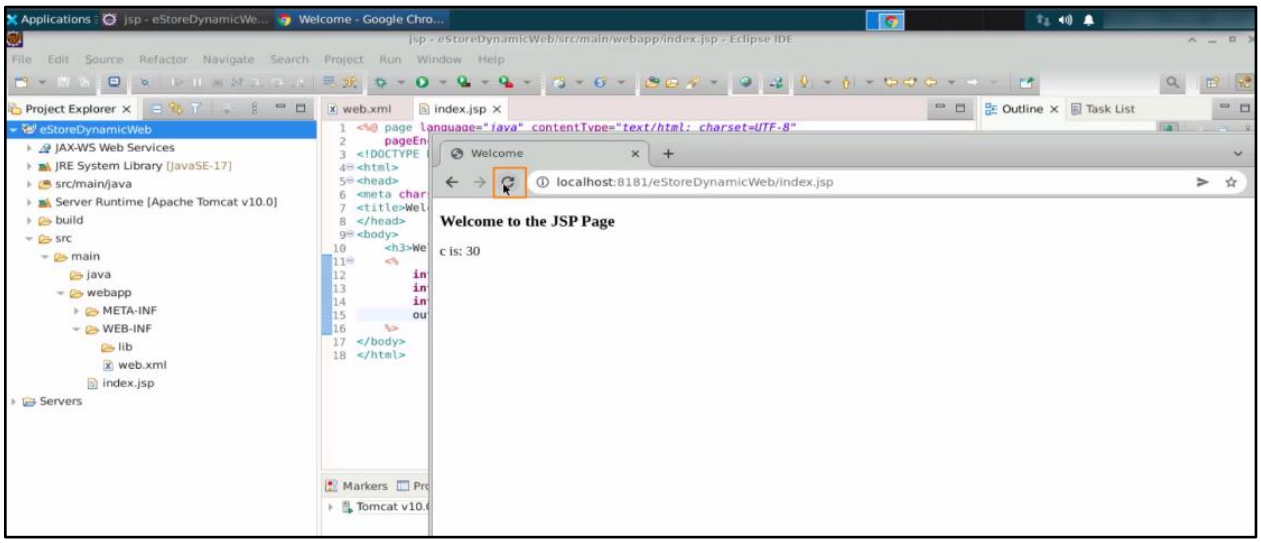
The JSP page will be displayed in the web browser.

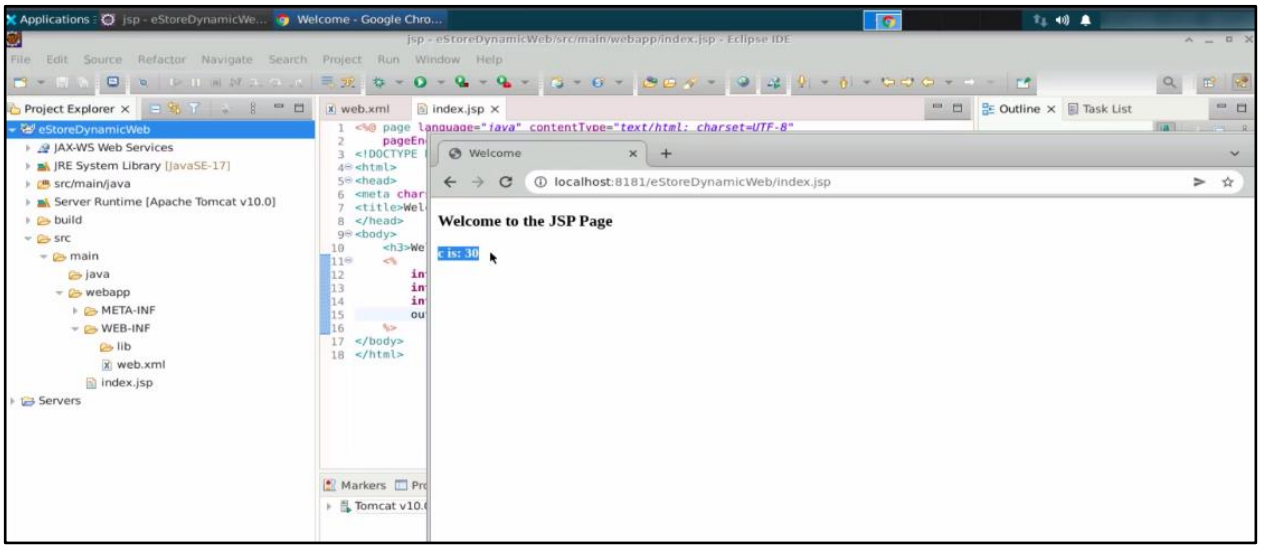
Step 3: Incorporate changes in the previous operation

3.1 If any changes are required, modify the necessary files or configurations based on the previous steps



3.2 Navigate back to the web browser where the JSP page is displayed and refresh the page

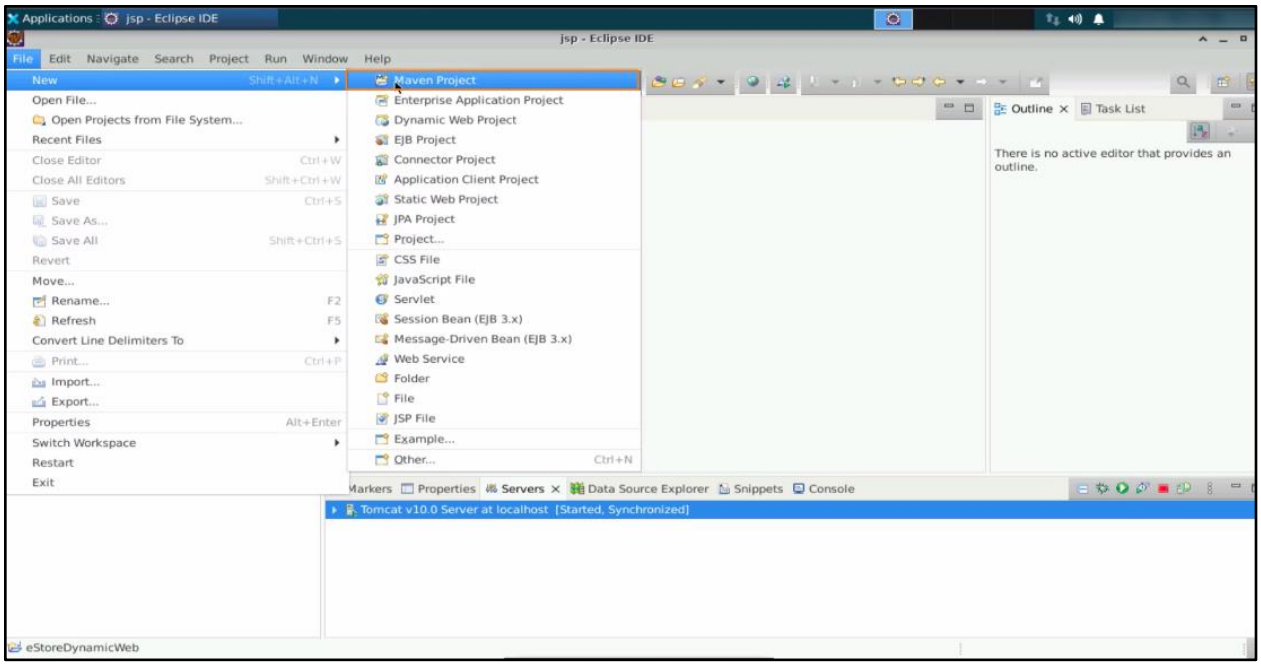




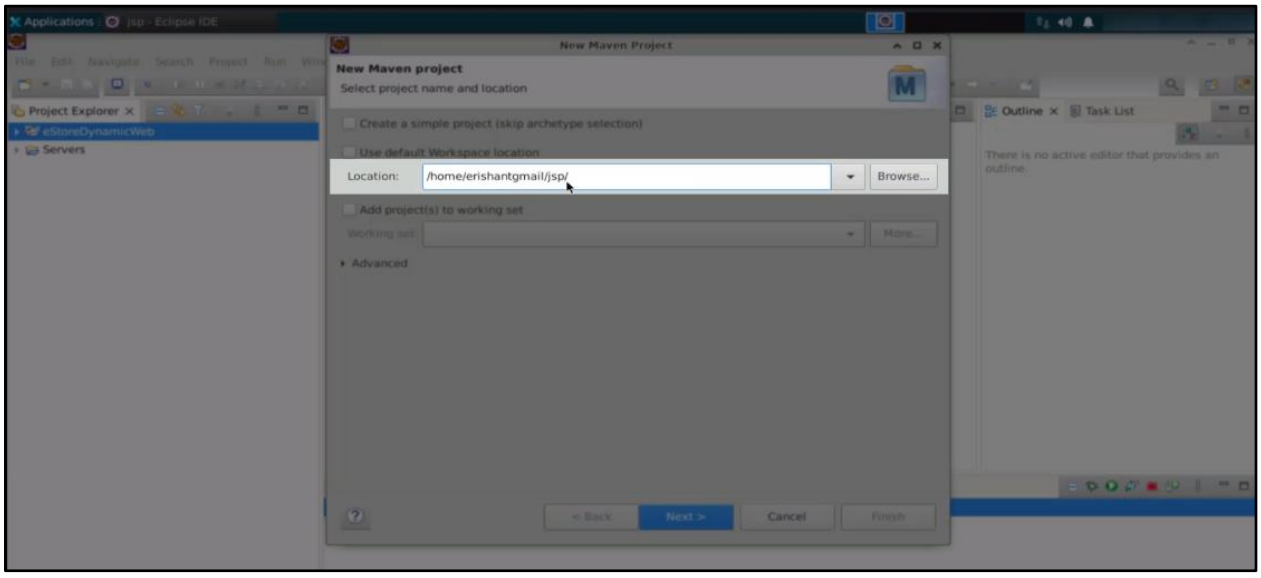
Observe the modifications made to the JSP page.

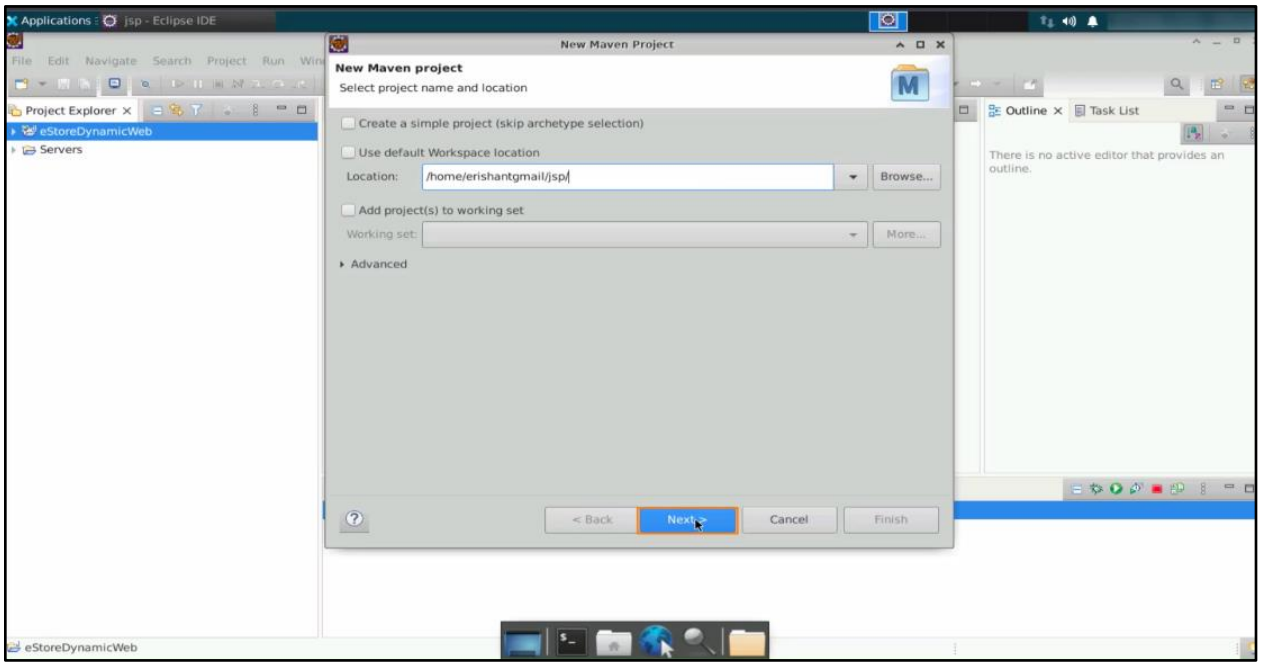
Step 4: Create a new Maven project

4.1 Create a new Maven project by following the path **File**->**New**->**Maven Project** in Eclipse

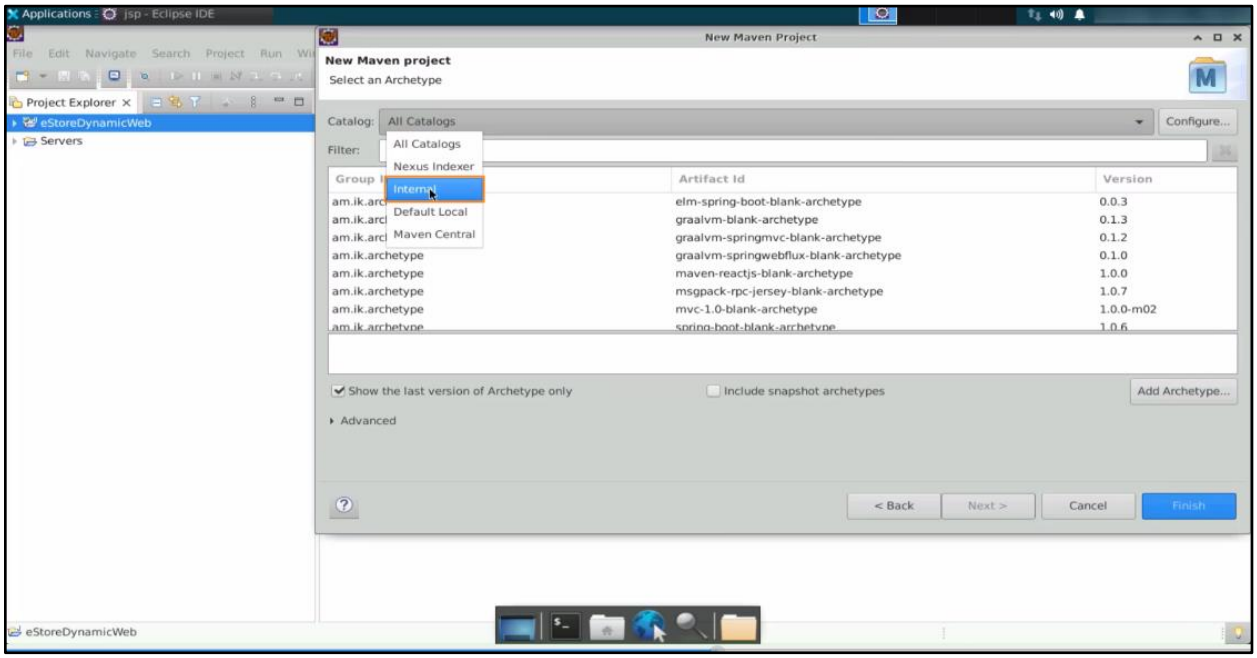


4.2 Choose a location for the project and click **Next**

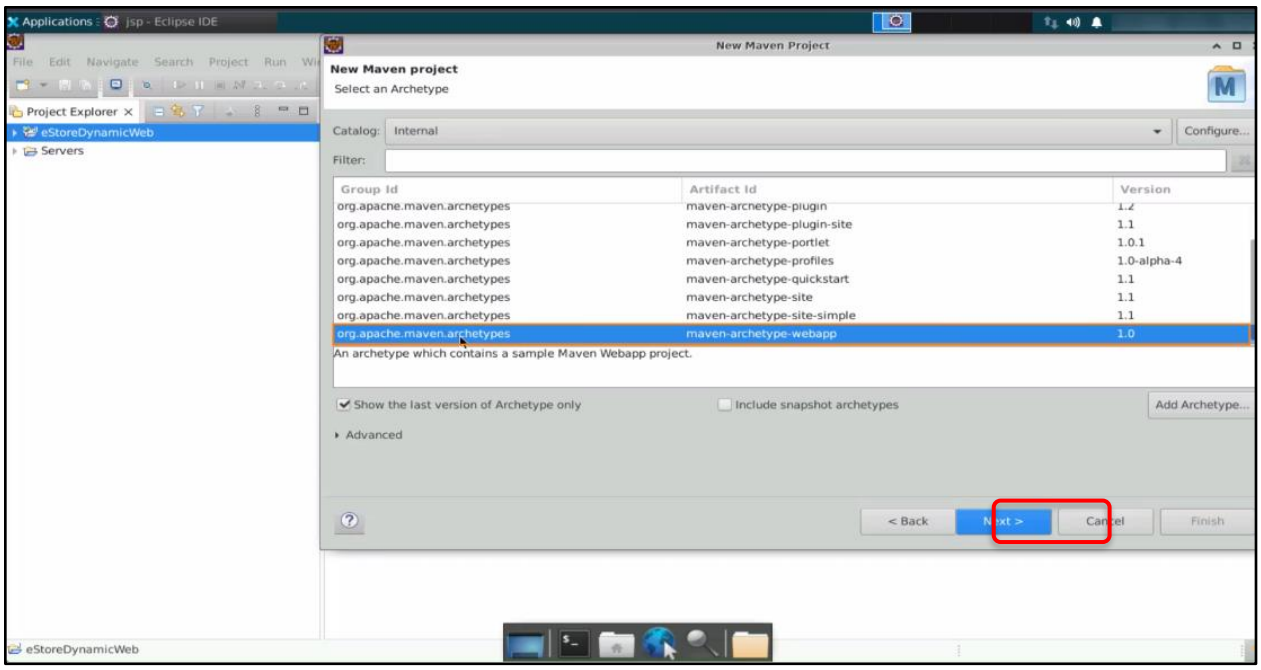




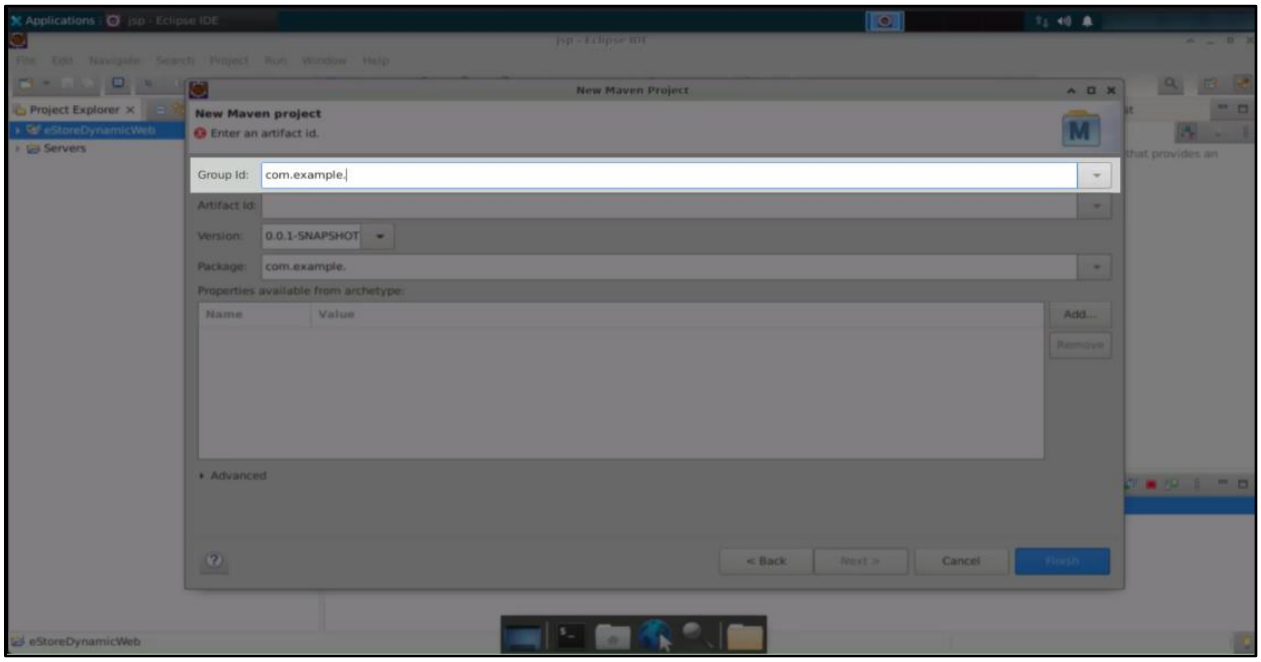
4.3 Select **Internal** as the **Catalog** and search for **maven-archetype-webapp**

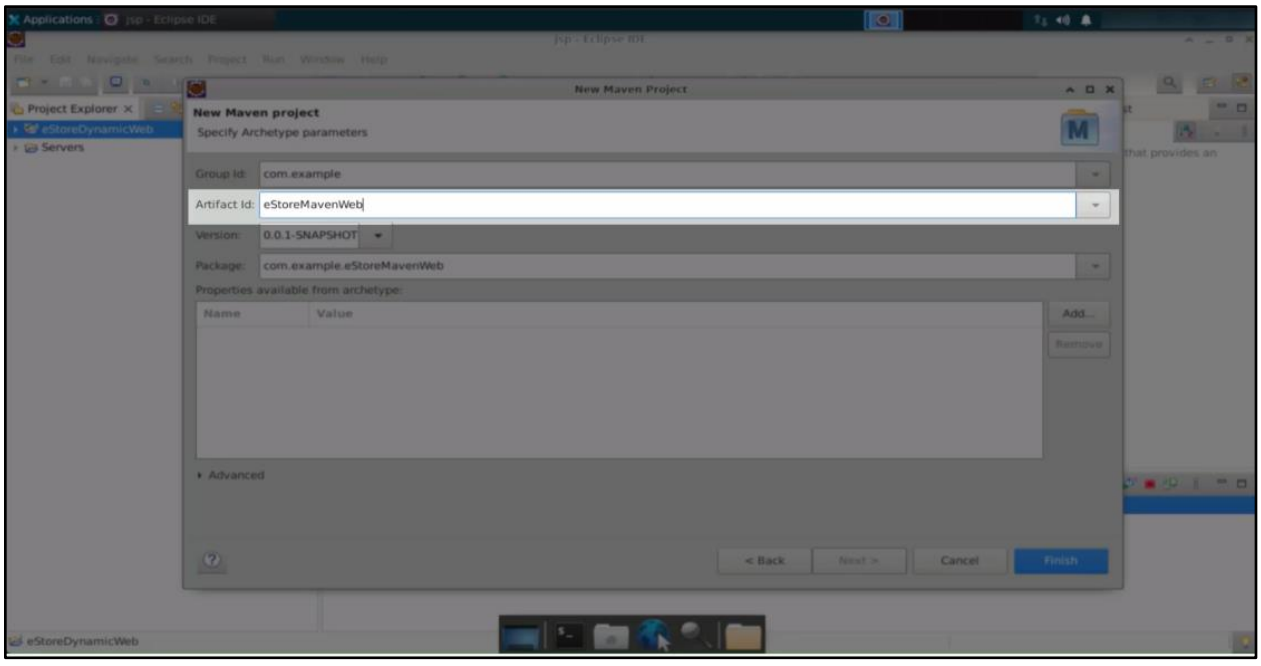


4.4 Select **maven-archetype-webapp** and click **Next**

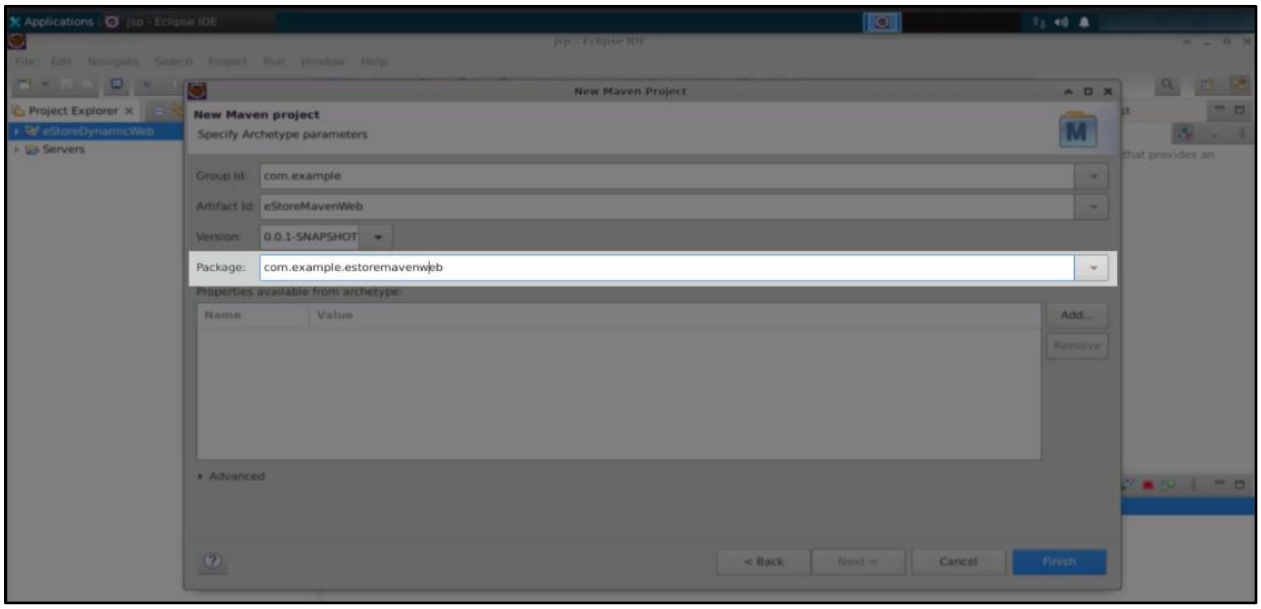


4.5 Set the Group Id as **com.example** and specify the Artifact Id as **eStoreMavenWeb**

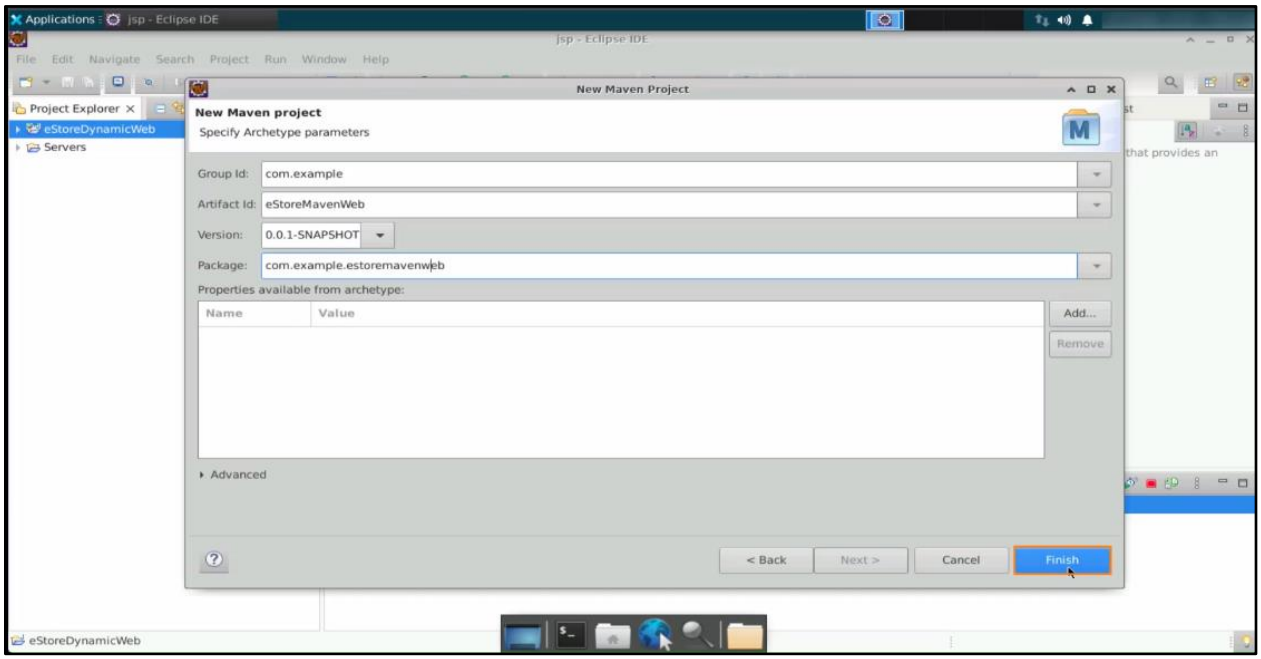




The package name will be automatically generated.

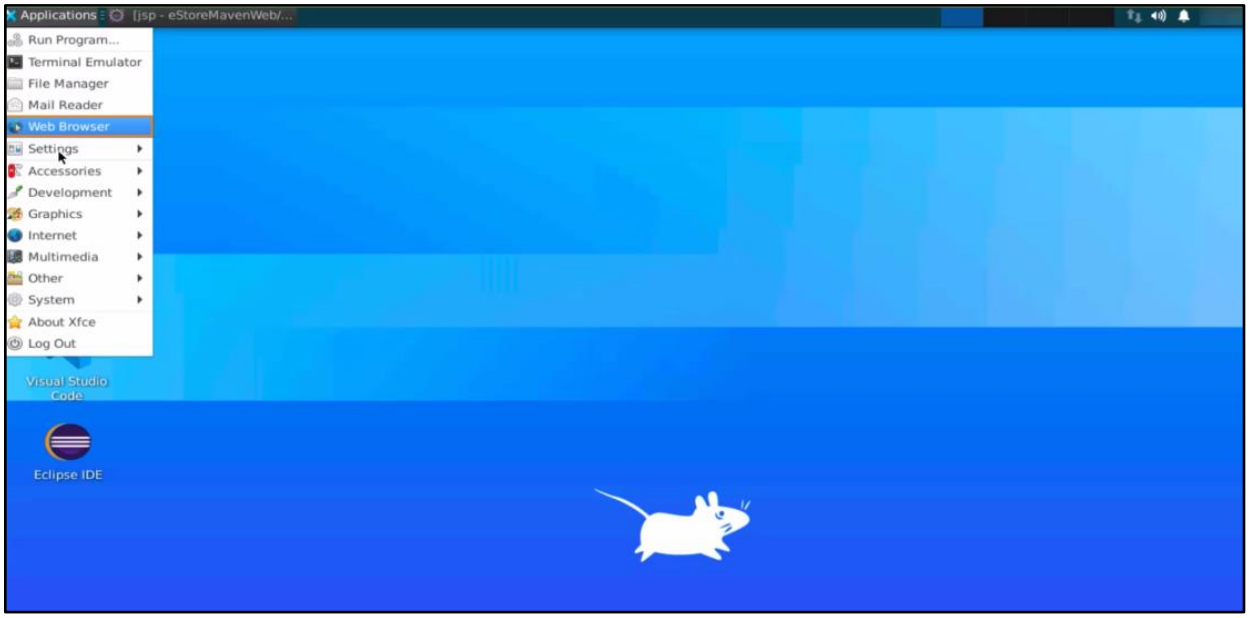


4.6 Click **Finish** to create the Maven project

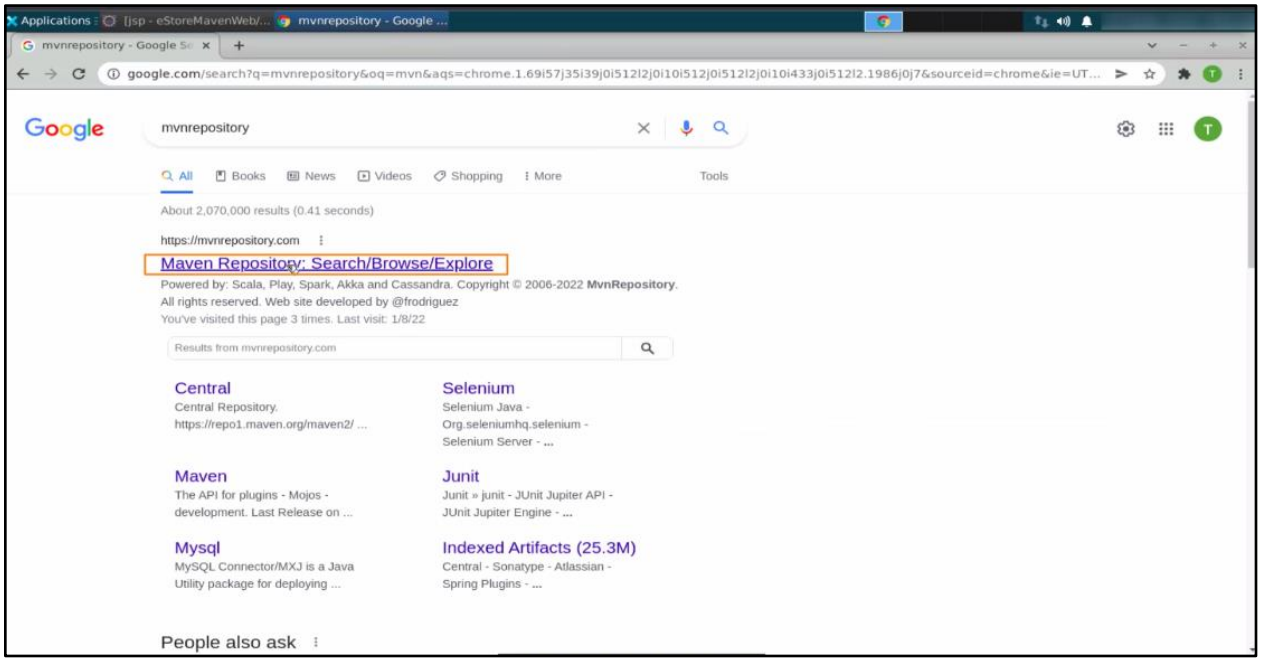


Step 5: Add the dependencies

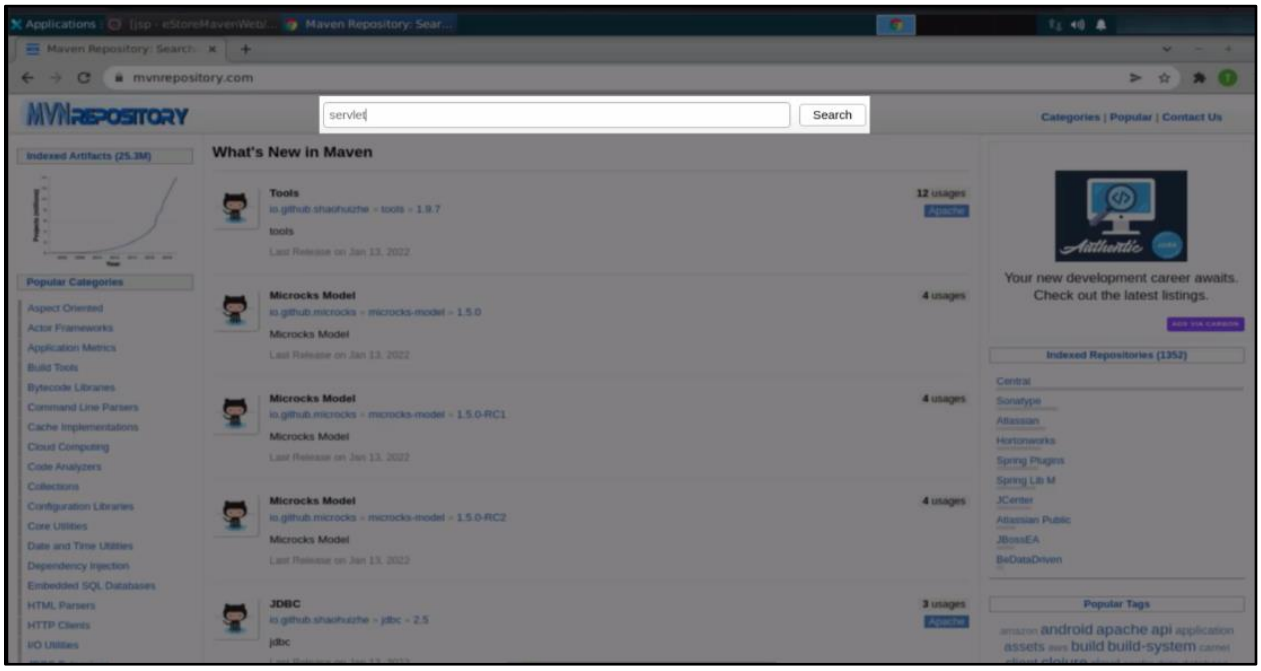
5.1 Launch a Web Browser



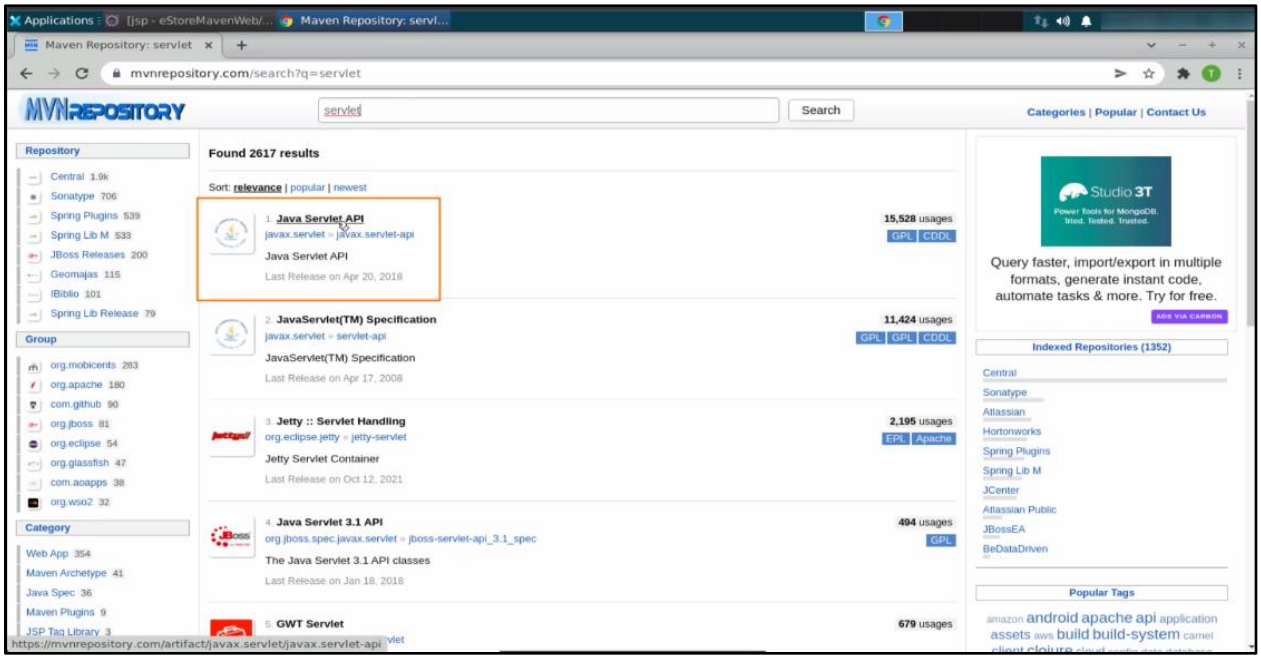
5.2 Search for **mvnrepository** and click on the search result for **mvnrepository.com**



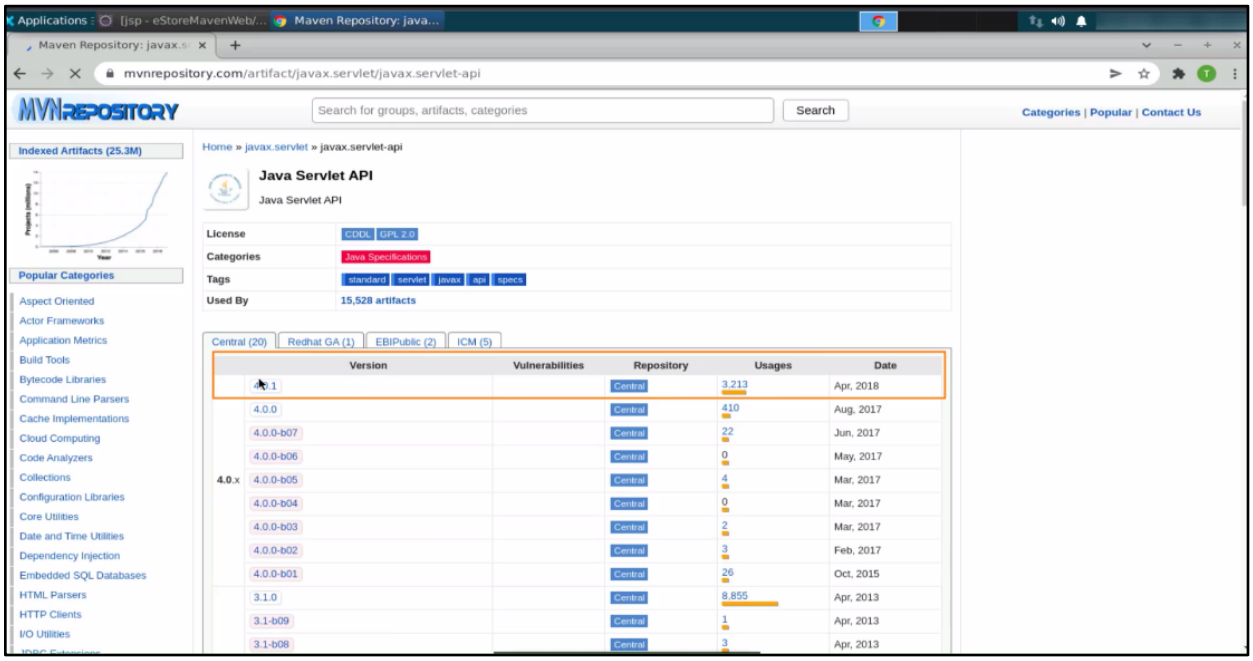
5.3 Search for servlet in the search bar on the website



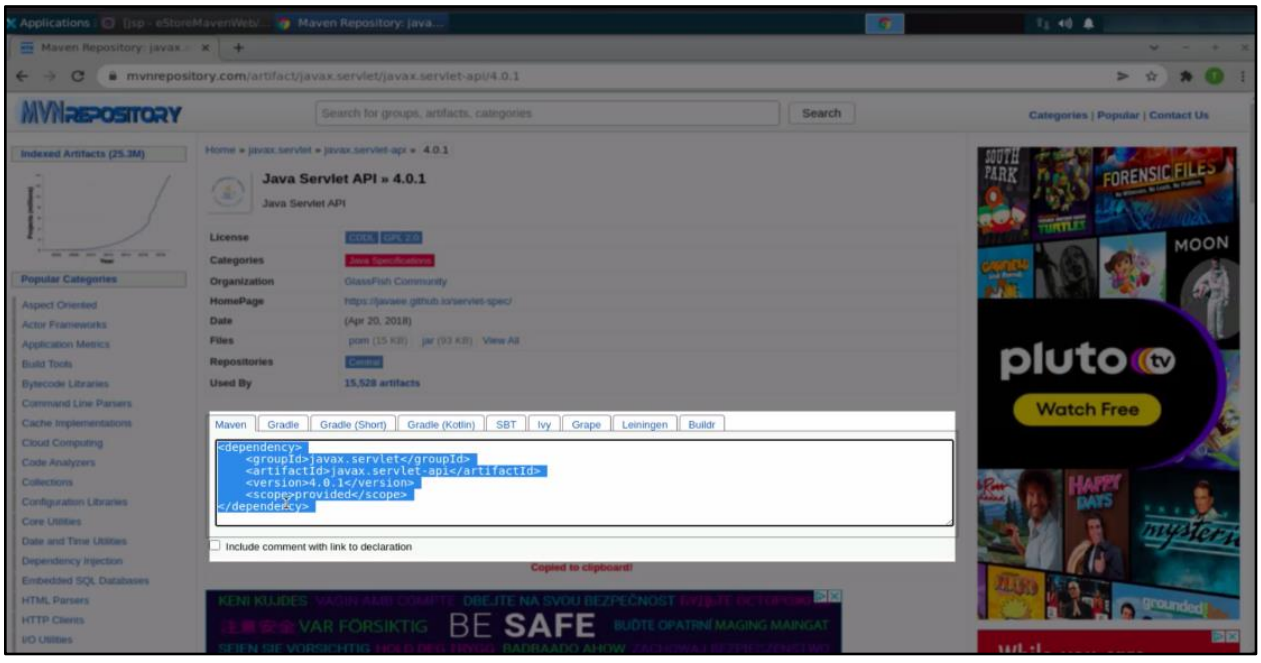
5.4 Open the **Java Servlet API** page from the search results



5.5 Select version **4.0.1** of the Java Servlet API

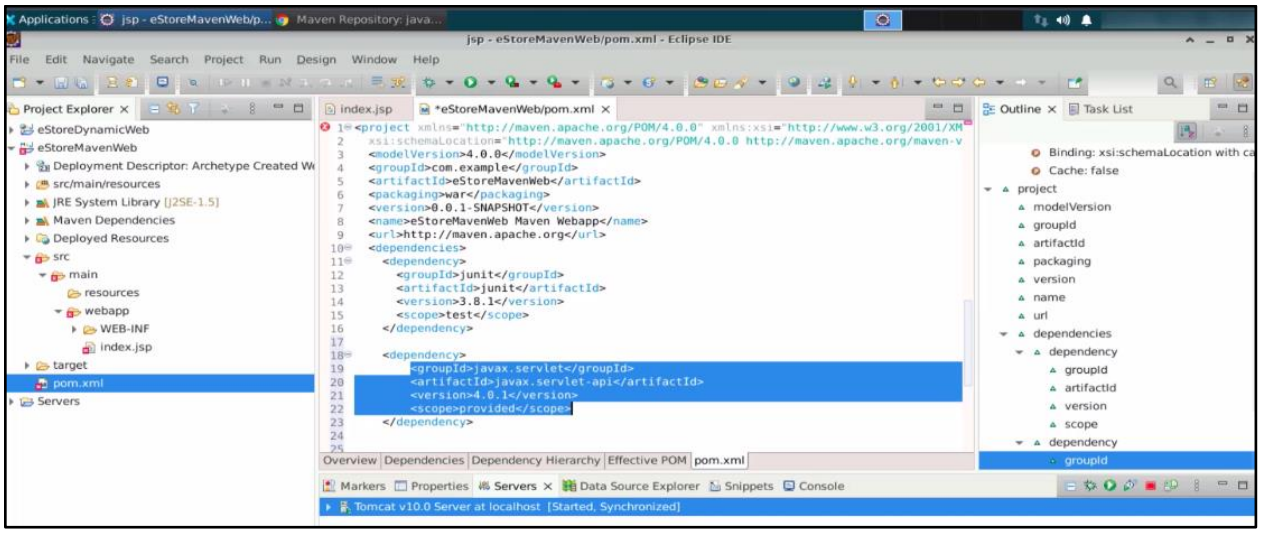


5.6 Copy the Maven dependency information provided on the page

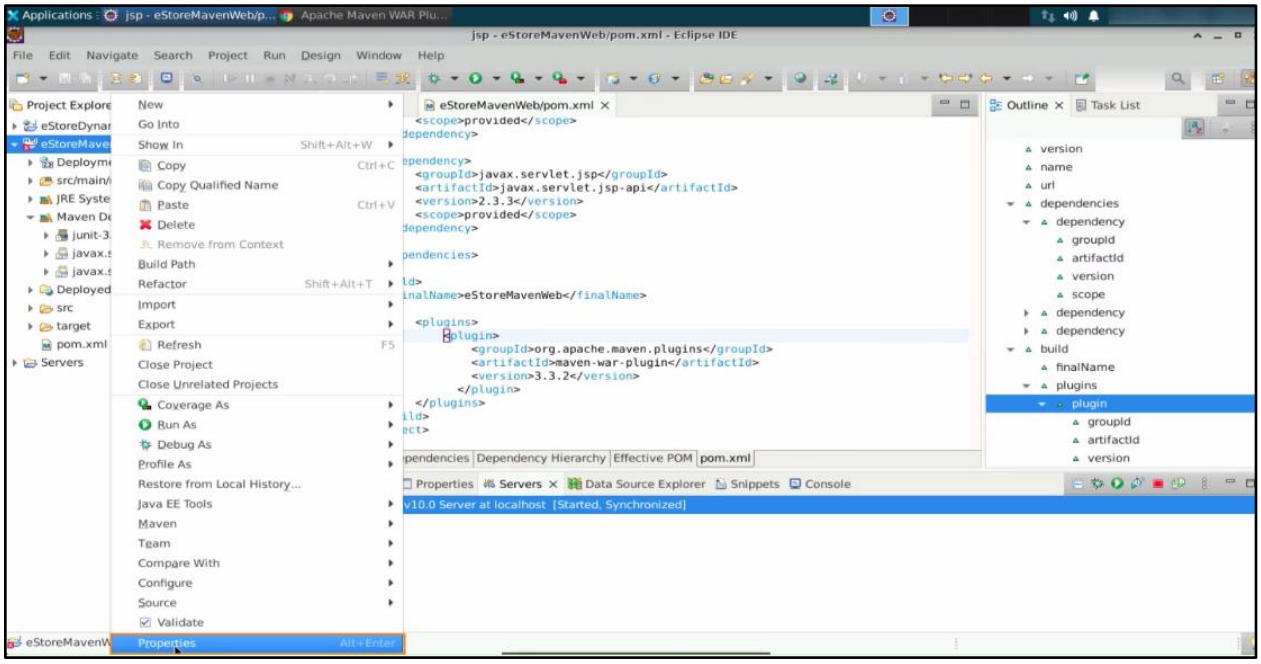


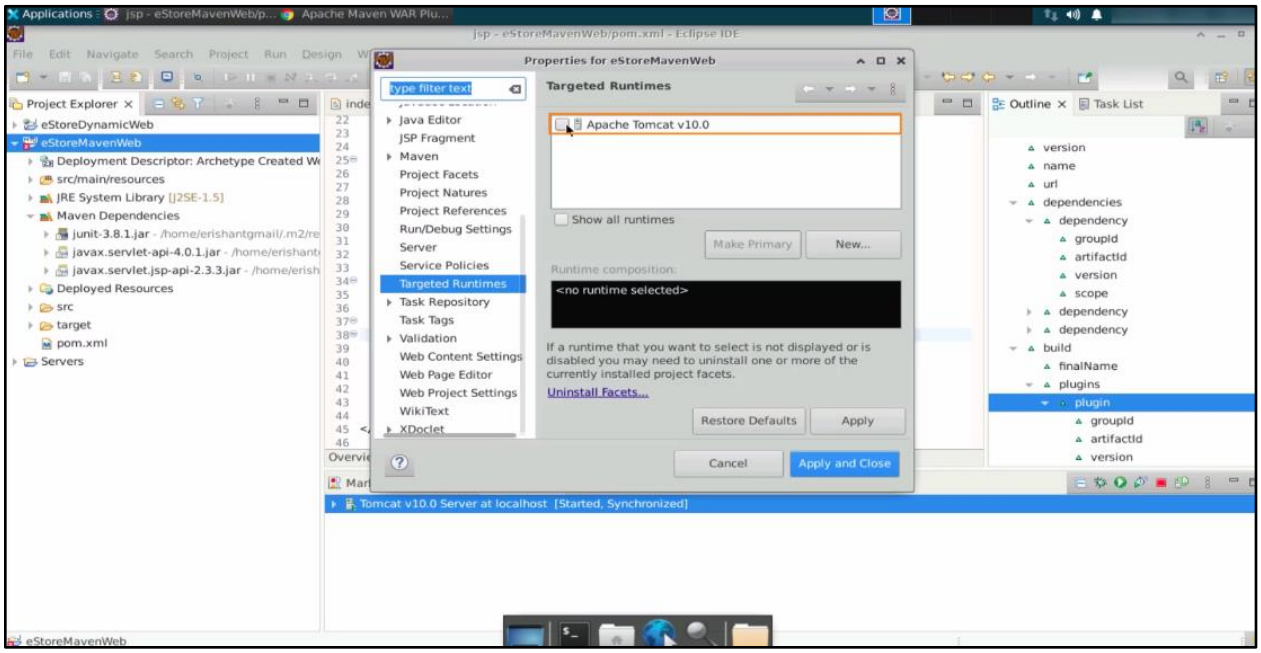
Return to the Eclipse IDE and open the **pom.xml** file in the Maven project

5.7 Paste the copied dependency information into the **pom.xml** file to include the Java Servlet API dependency

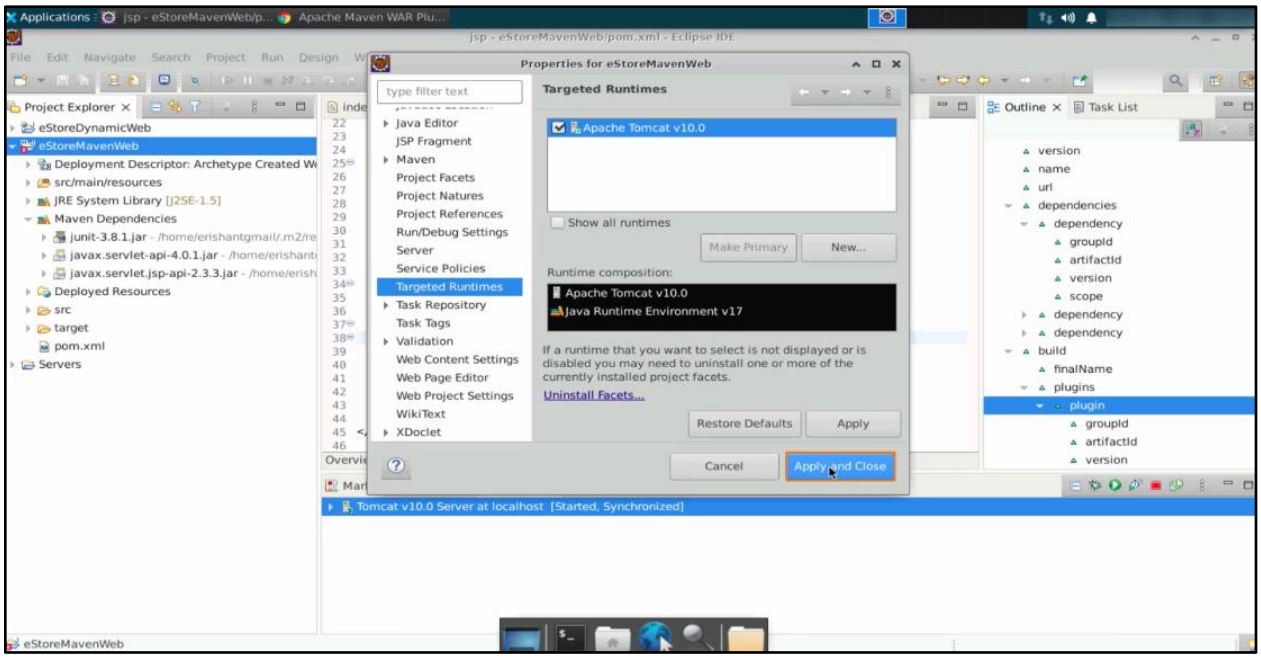


5.8 Go to the project properties by right-clicking on the project, selecting **Properties**, and choosing **Targeted Runtimes**



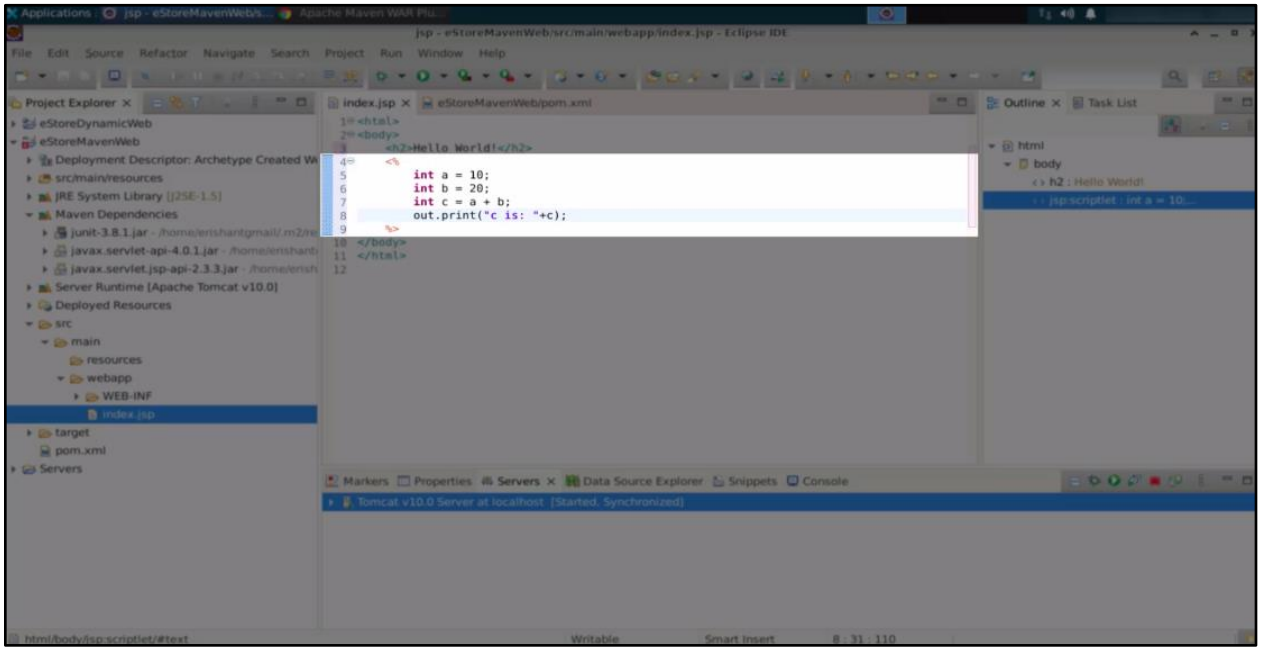


5.9 Check the box for the Apache Tomcat runtime and click **Apply and Close** to configure the targeted runtime

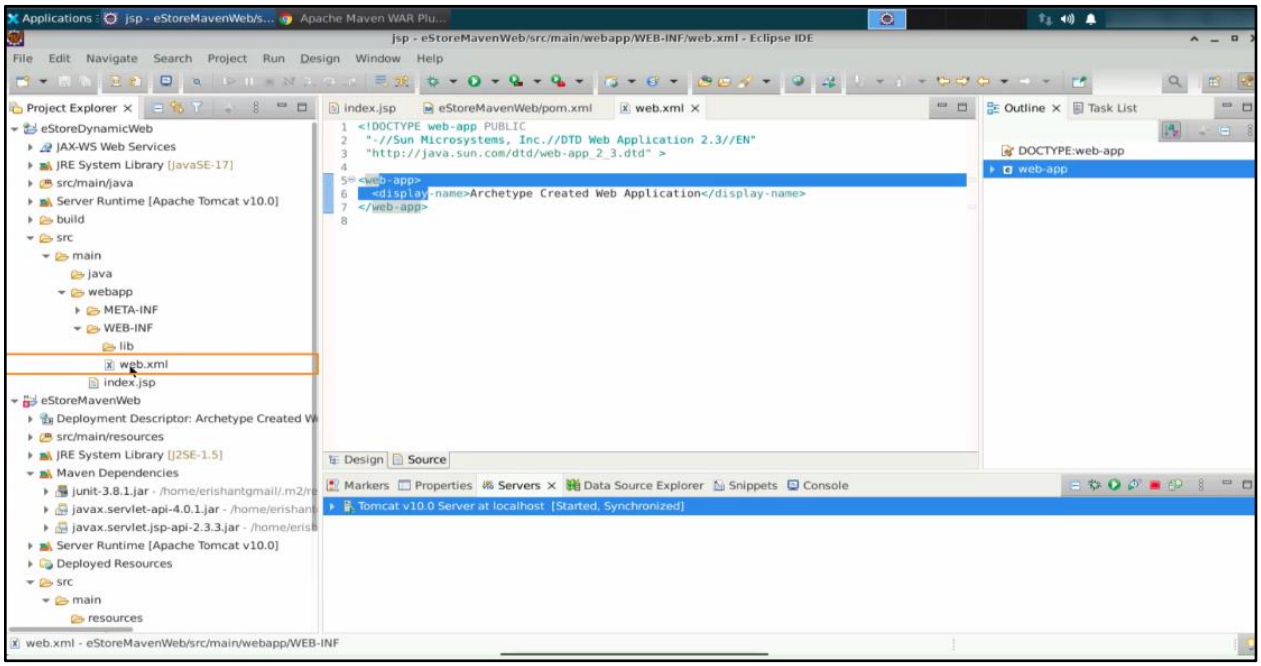


Step 6: Create a scriptlet tag

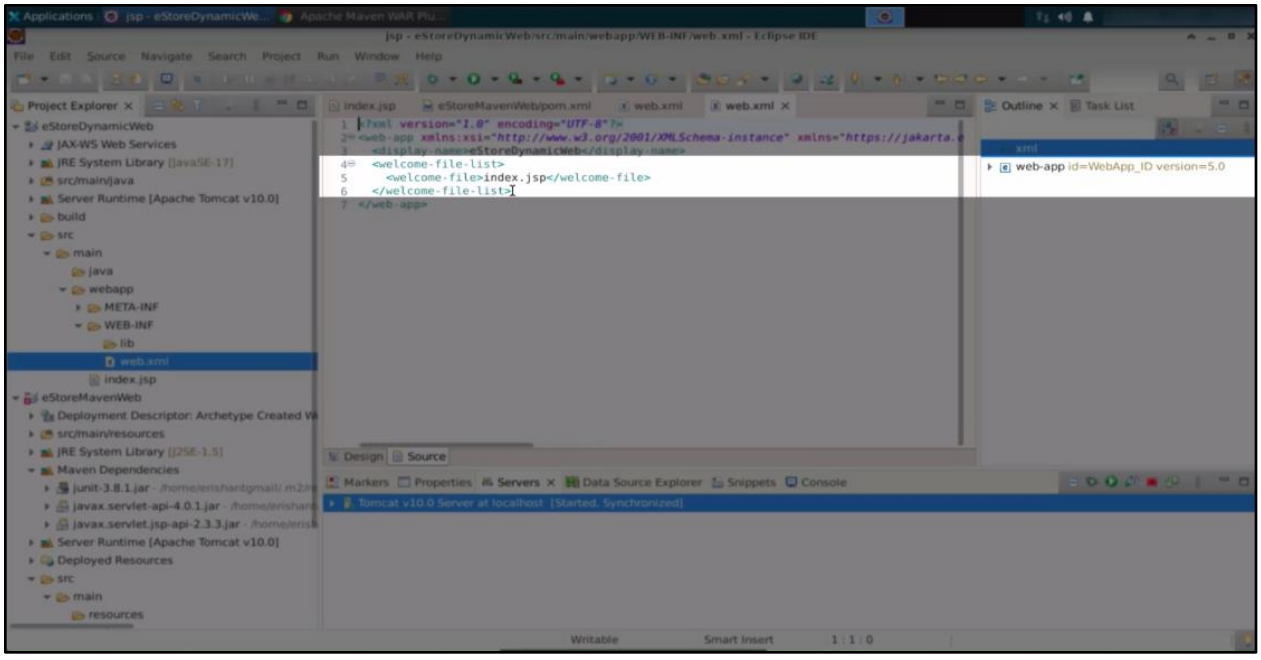
6.1 Go to the **index.jsp** page and create a scriptlet tag **<% %>** to add server-side logic or code



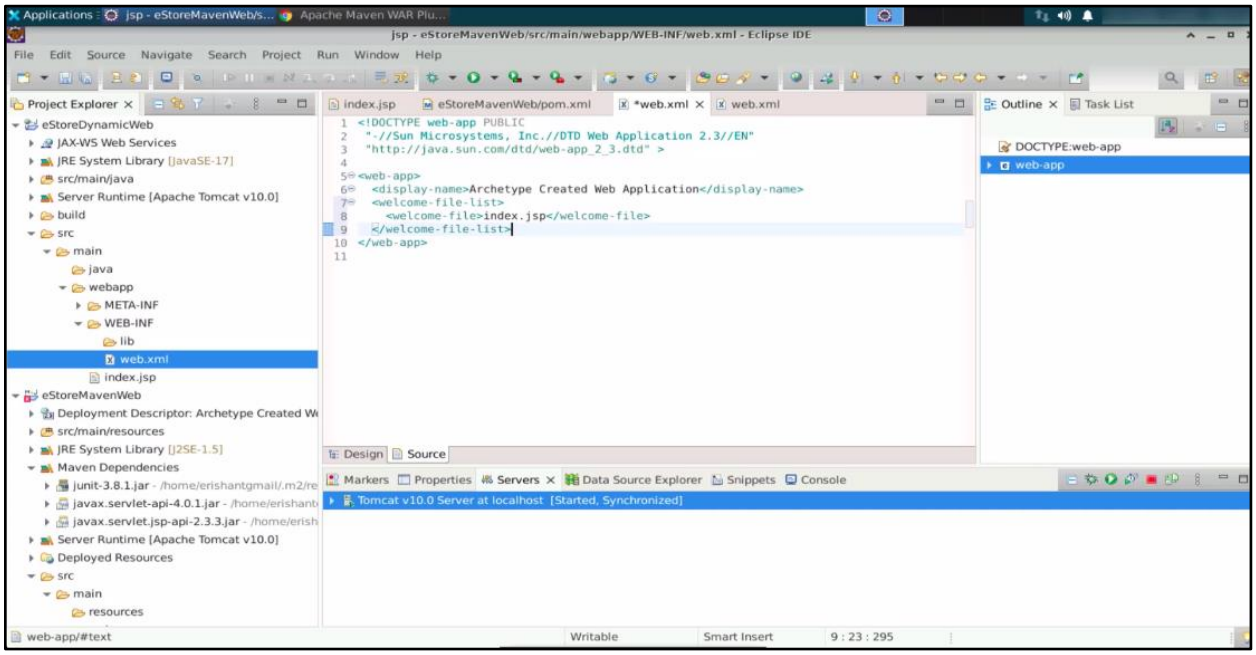
6.2 Open the **web.xml** file located in the **WEB-INF** directory



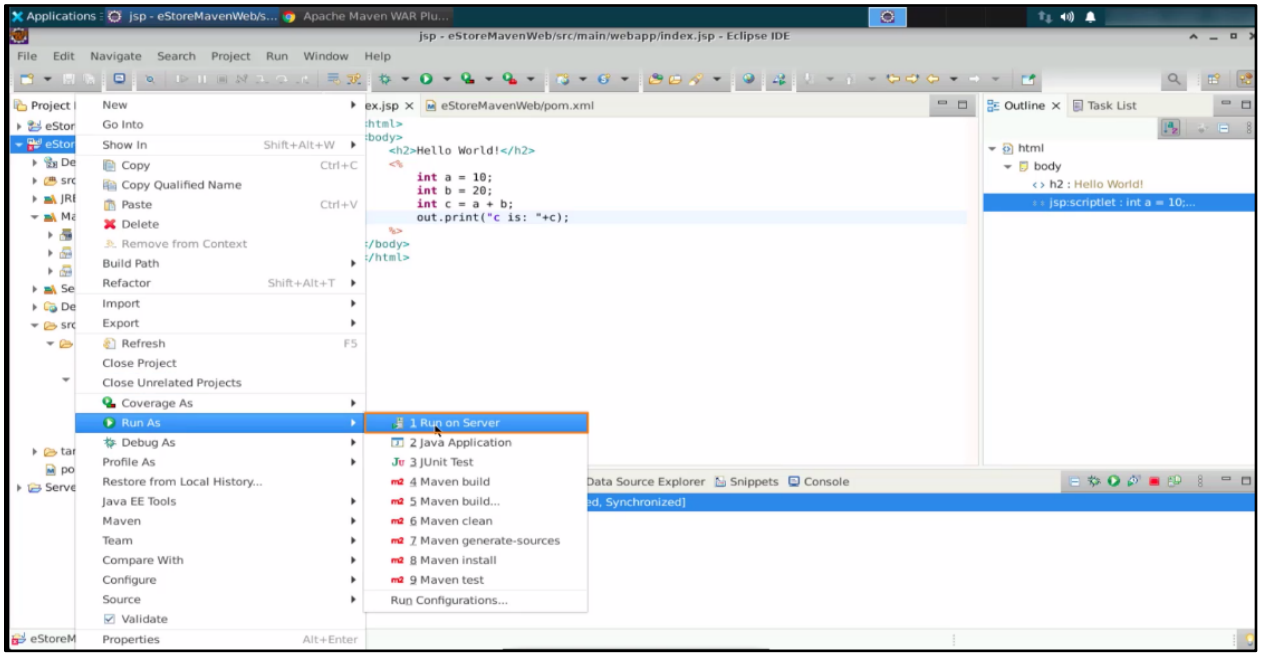
6.3 Copy the necessary code or configuration from the documentation or example



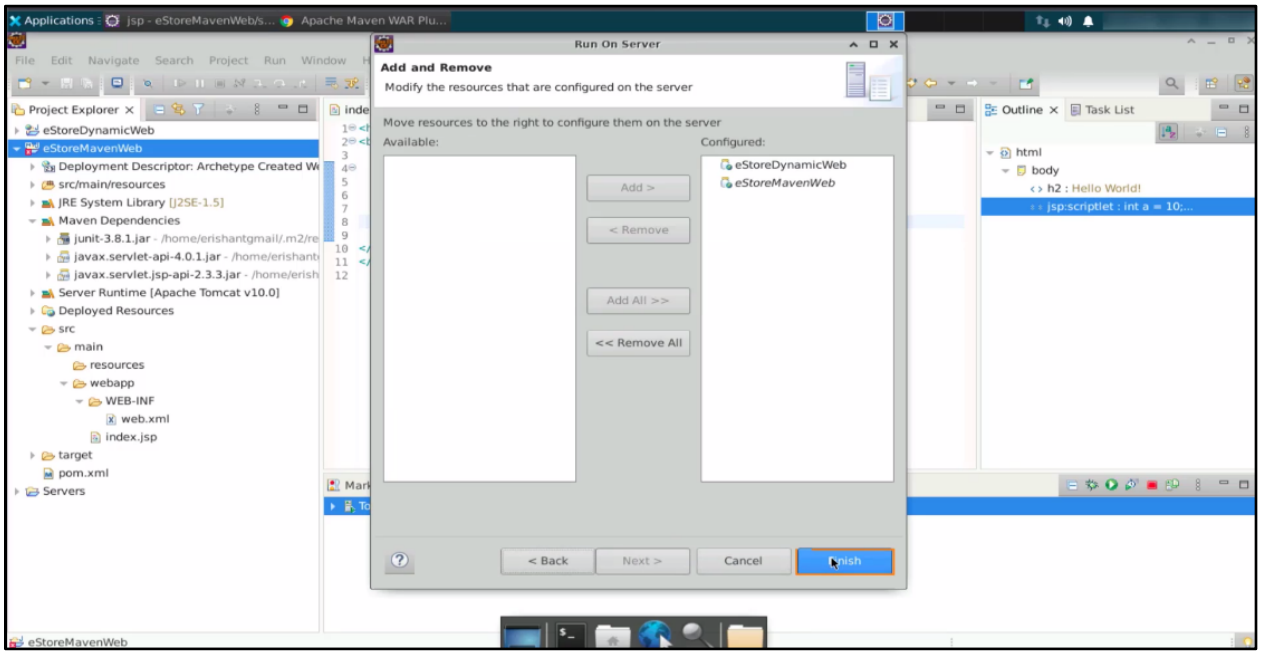
6.4 Paste the copied code or configuration into the **web.xml** file and save it



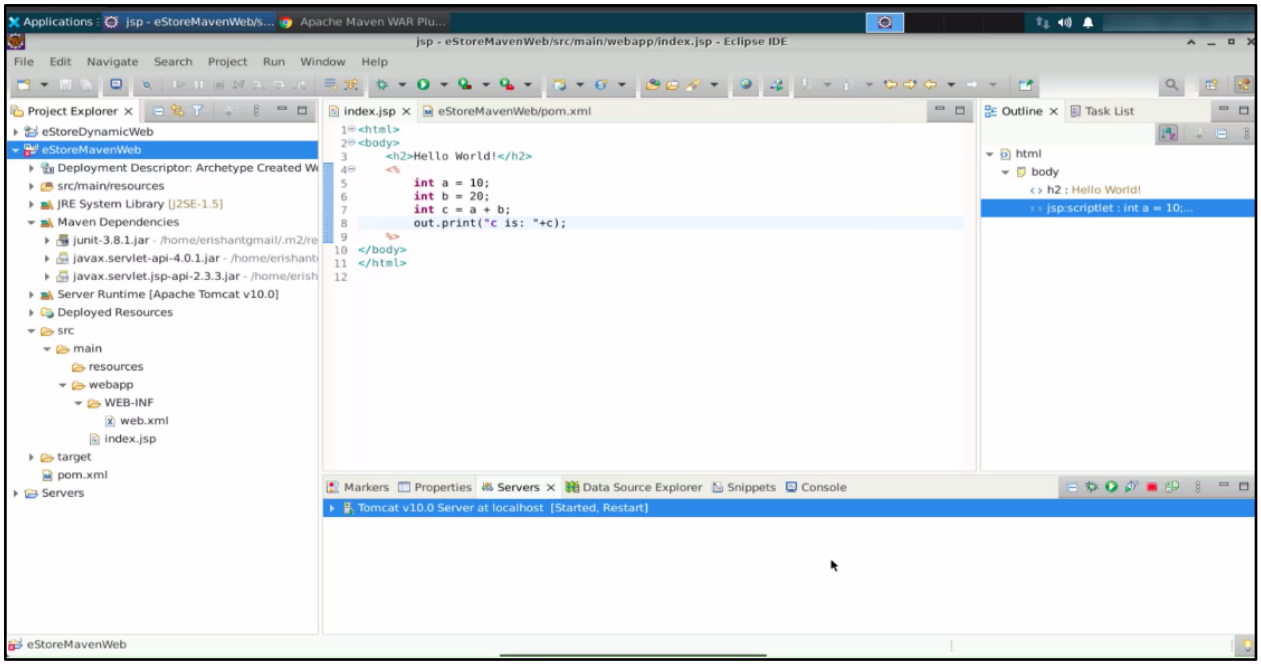
6.5 Right-click on the project, select **Run As**, and click **Run on Server** to run the project again



6.6 Choose the configured Tomcat server and click **Finish** to run the project on the server



Observe the changes made and displayed in the web browser.



Following these steps, you have successfully created a JSP (Java Server Pages) page using Eclipse IDE.