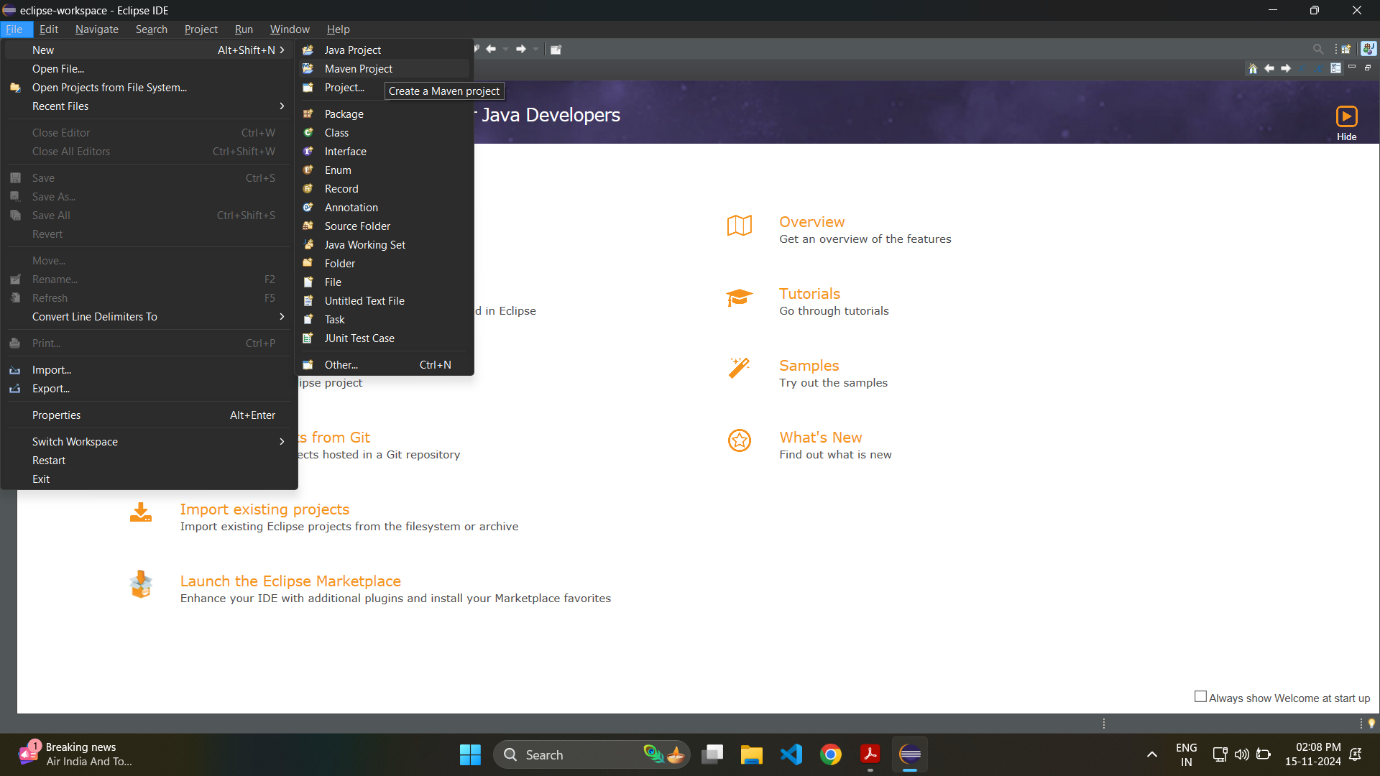
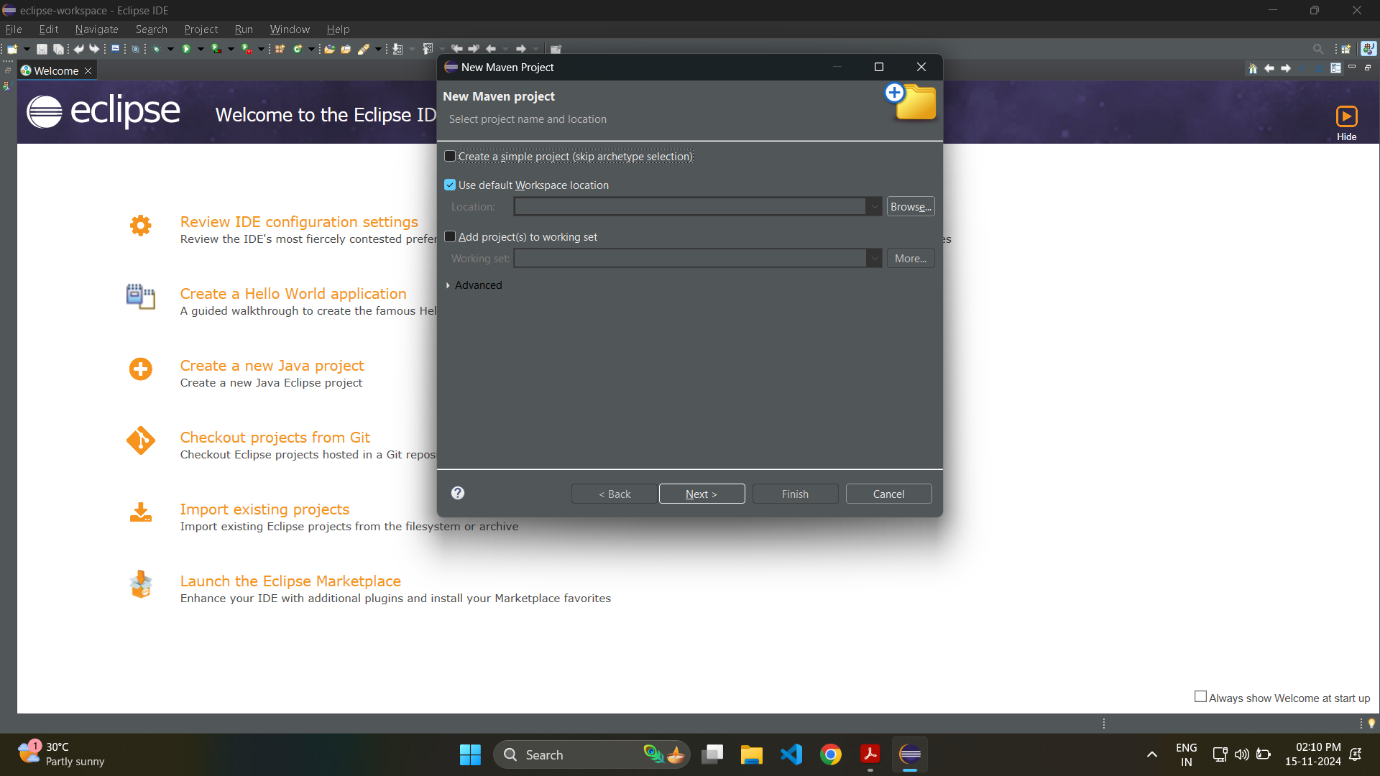
**Exercise 5B: Creating a new Maven Project**

N Anand Sai (22BD1A6615)

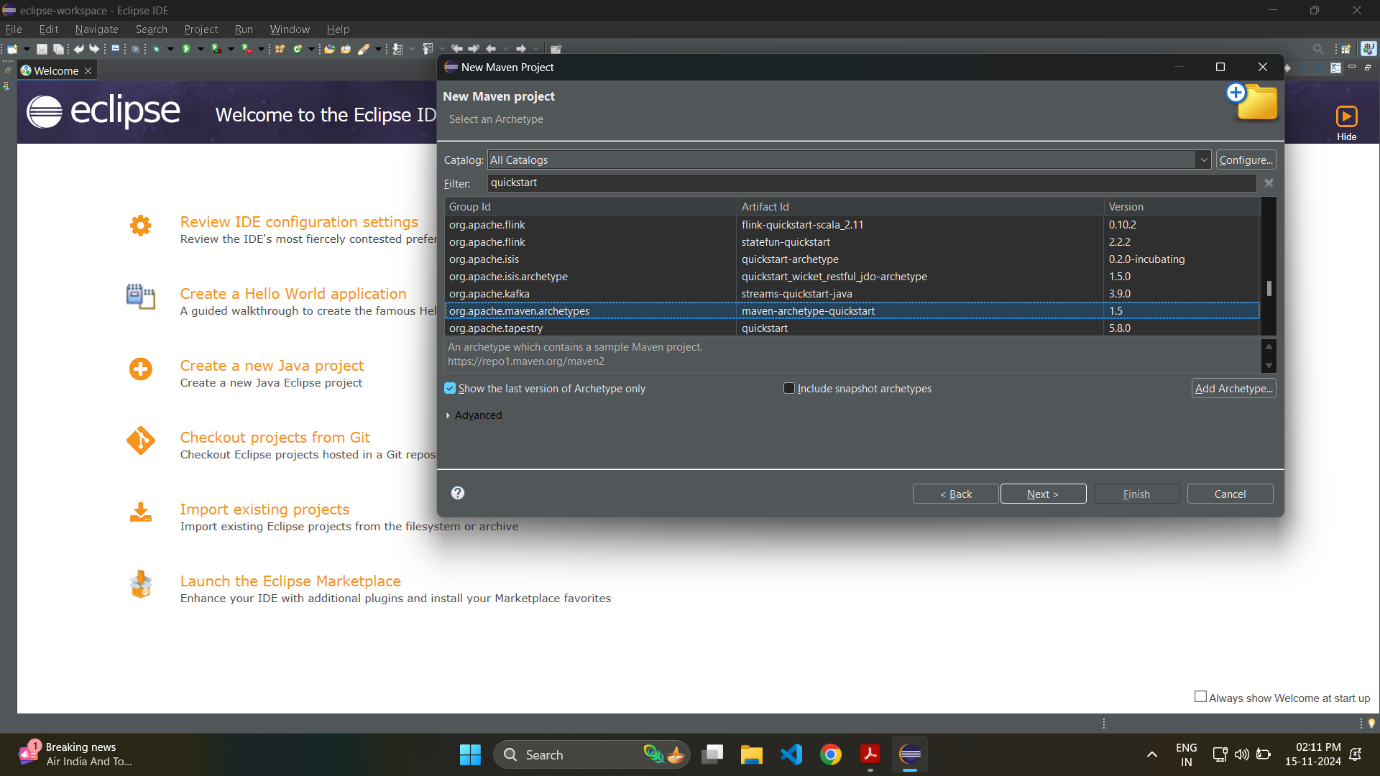
Step 1: In Eclipse, click on File -> New -> Maven Project.



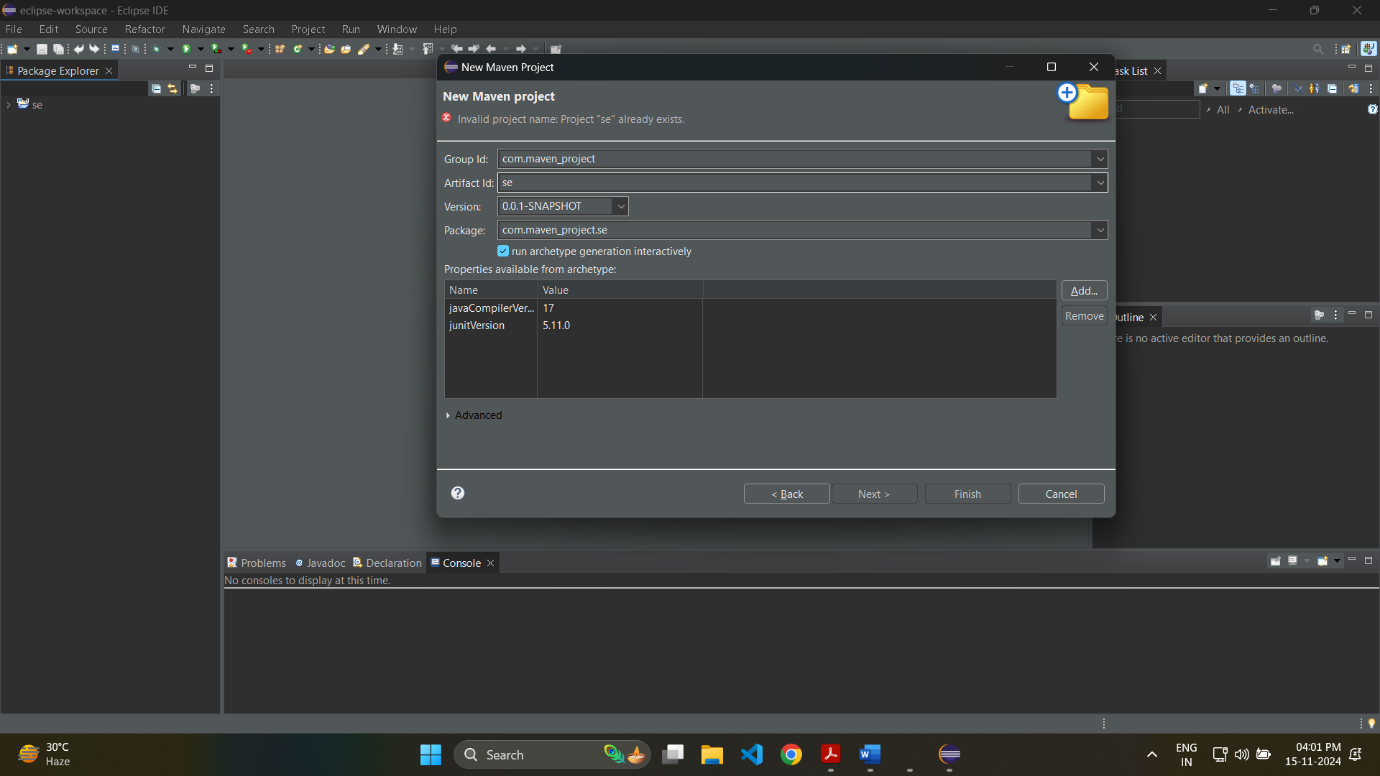
Step 2: Make sure “Use default Workspace location” is checked and click on “Next”.



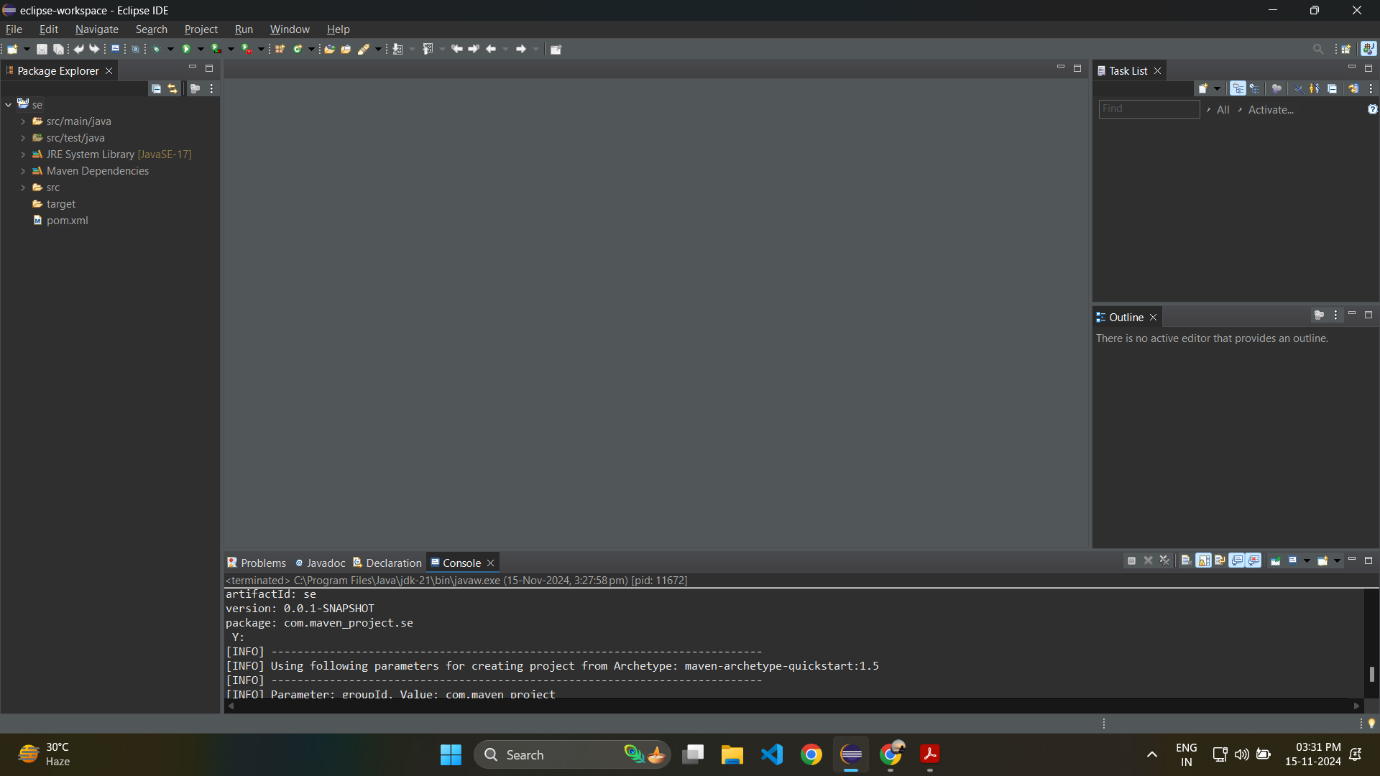
Step 3: When selecting archetype, enter “quickstart” in filter and select “org.apache.maven.archetypes” in the menu below. Click on “Next”.



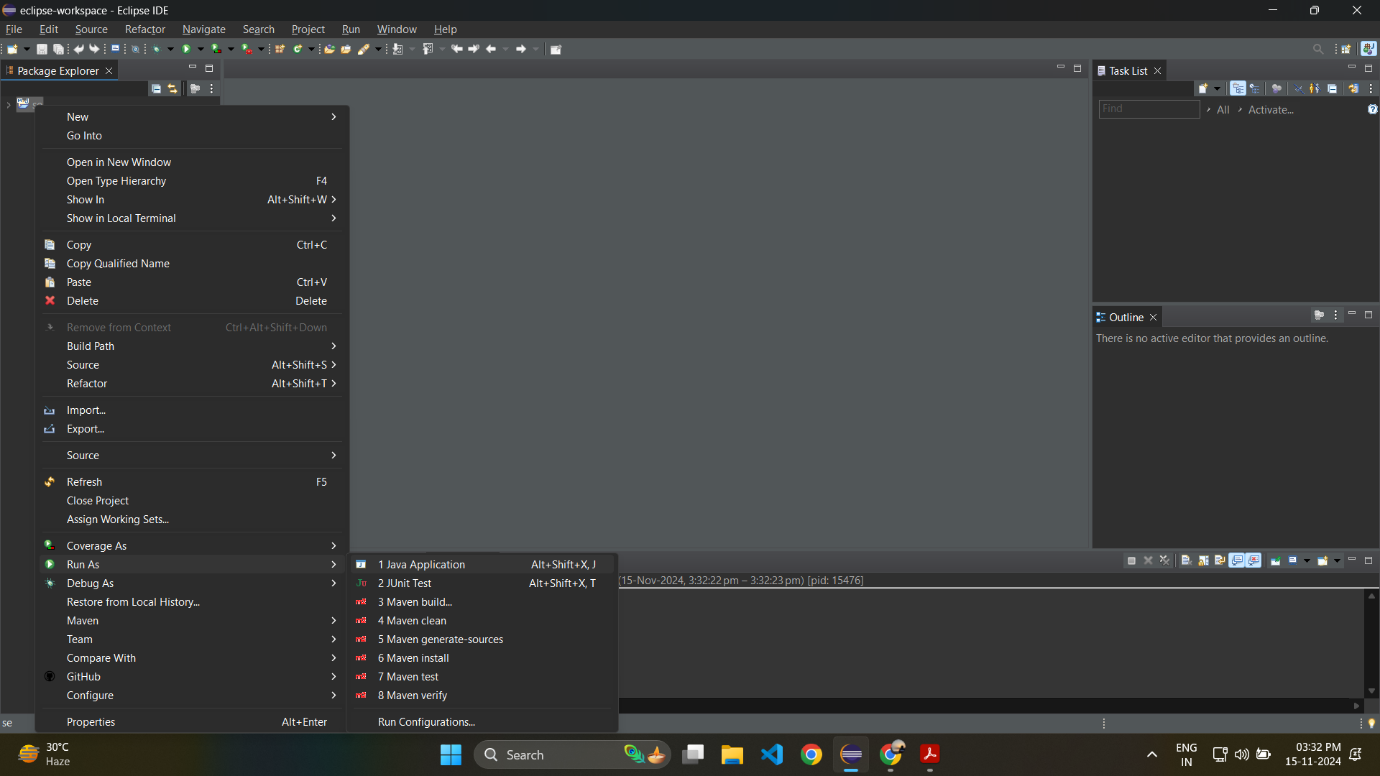
Step 4: Enter group id in the format com.project\_name and an artifact Id and press on “Finish” to create the project.



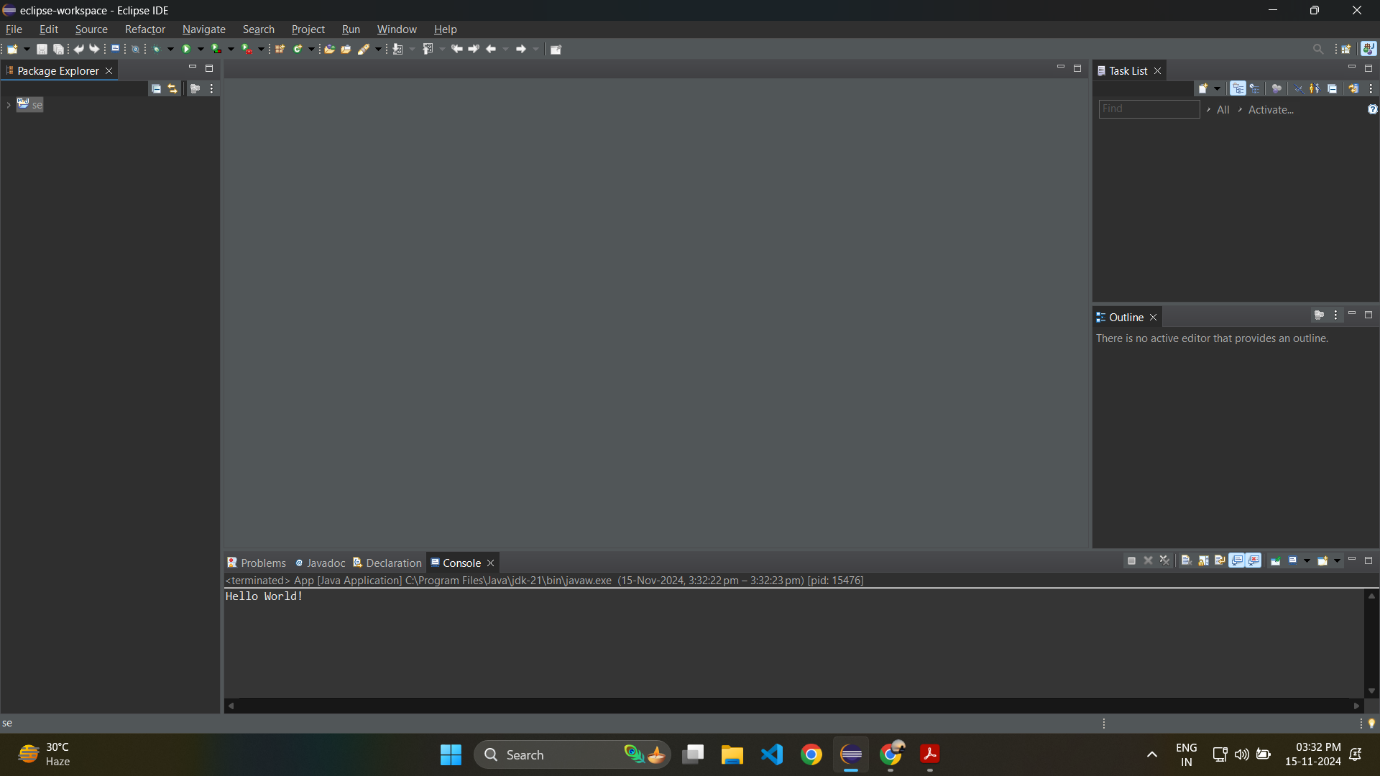
Step 5: You can check the progress of the download in the Console. You can see the project files after the project is ready.



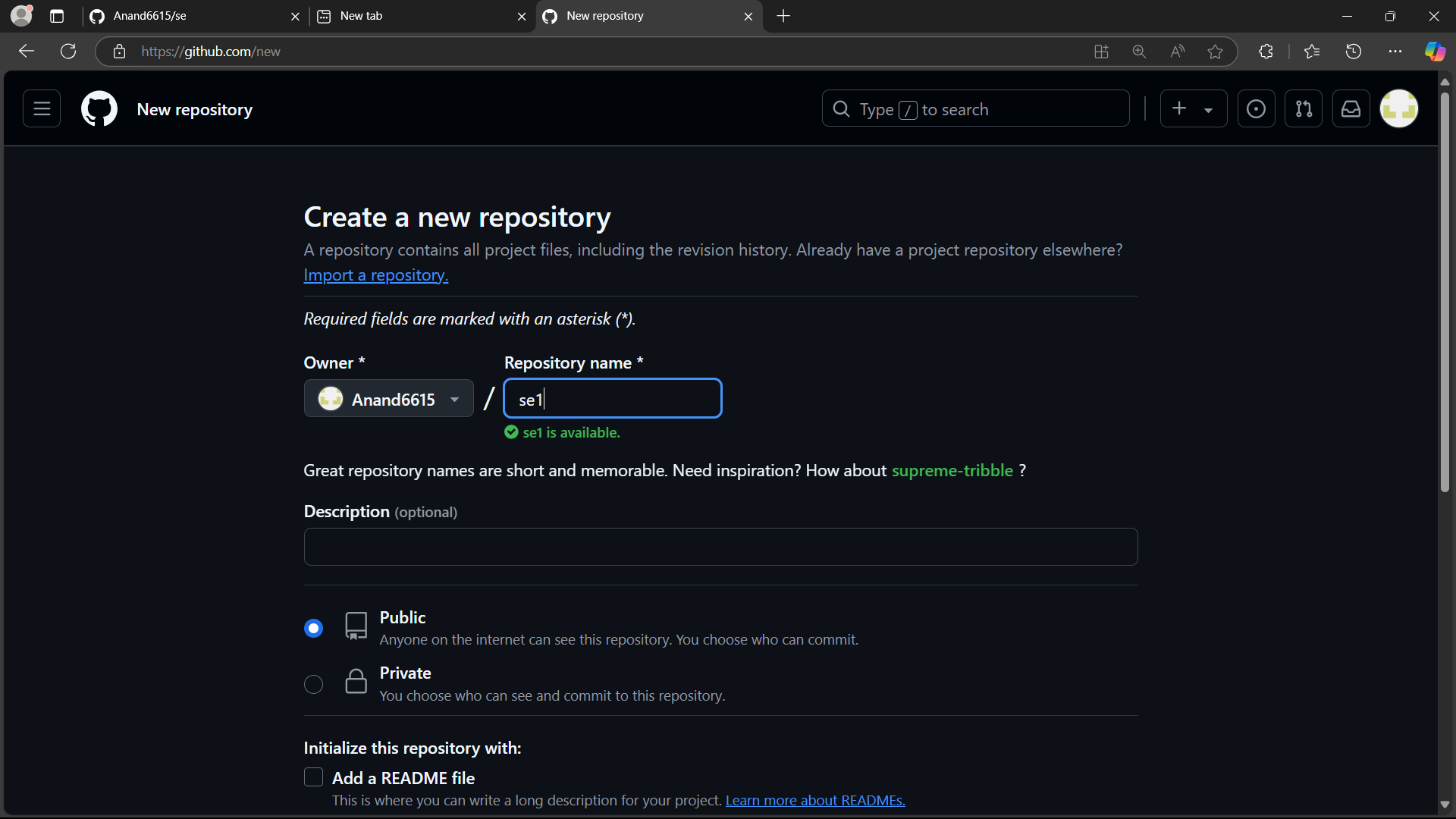
Step 6: After the project is ready, right-click on the project, Run As -> Java Application.



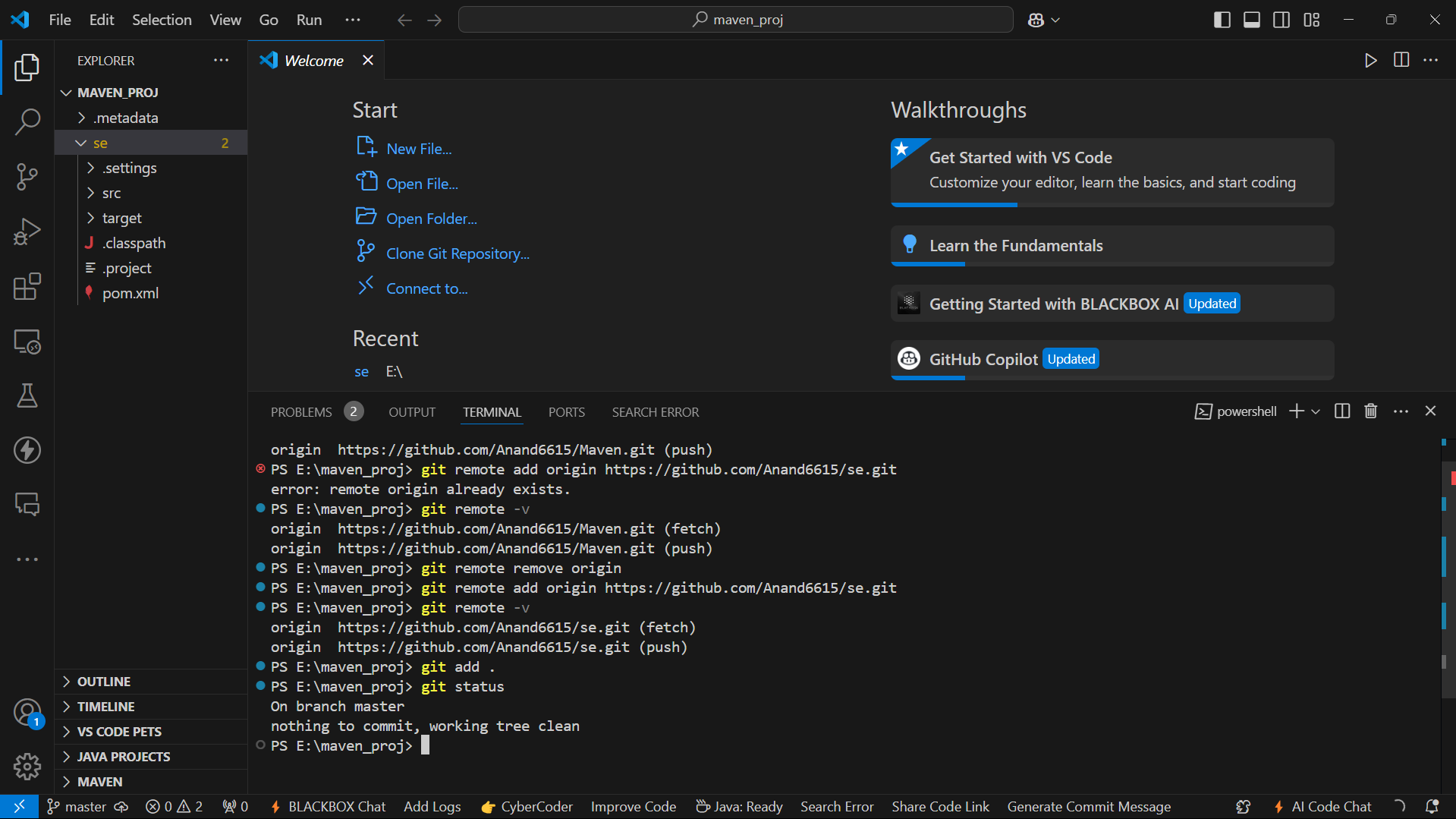
Step 7: You can see “Hello World” in the console if everything works fine.

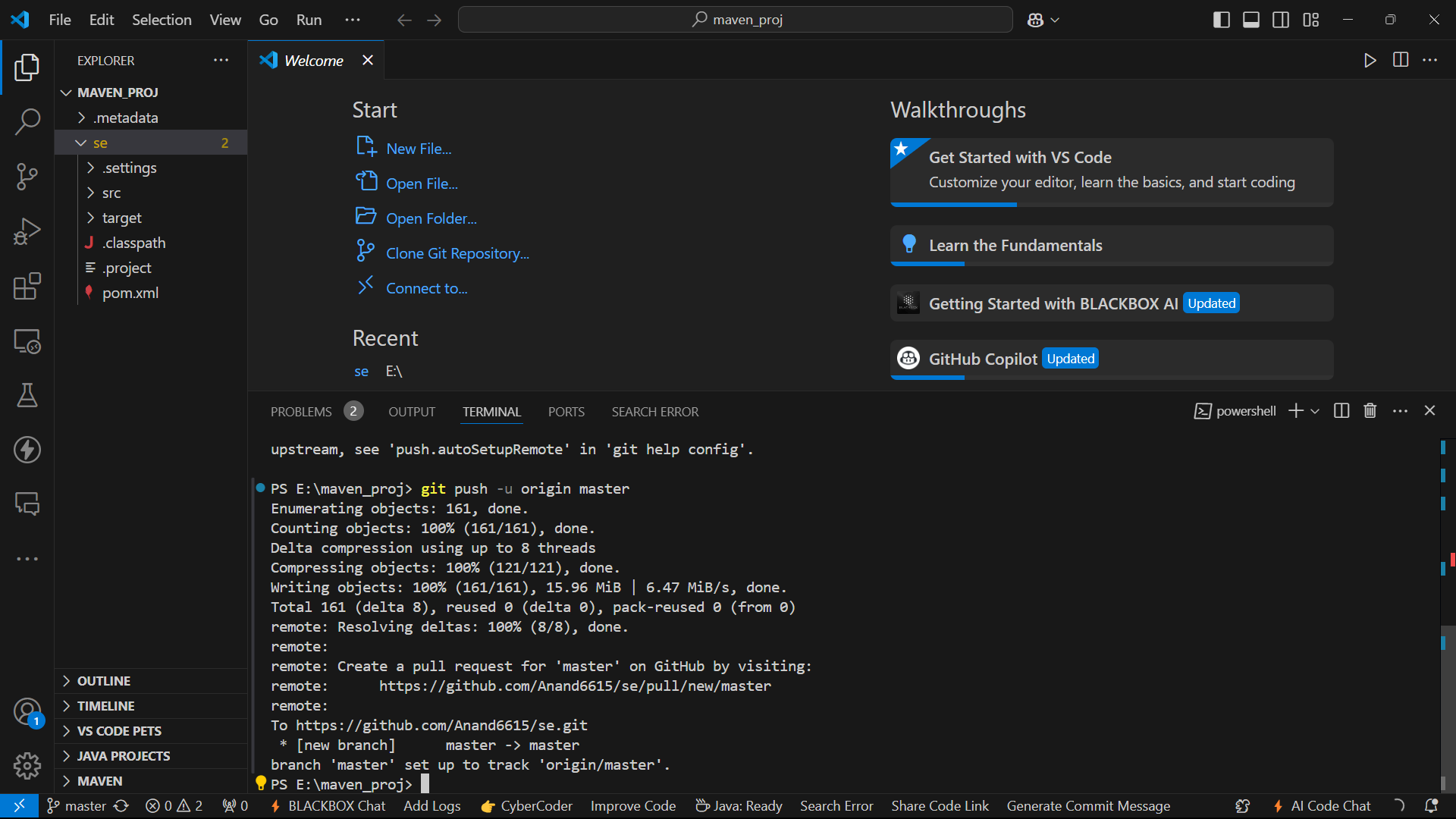


Step 8: To push the code onto GitHub, create a new GitHub repository. Select .gitignore template as Maven, and click on “Create”.

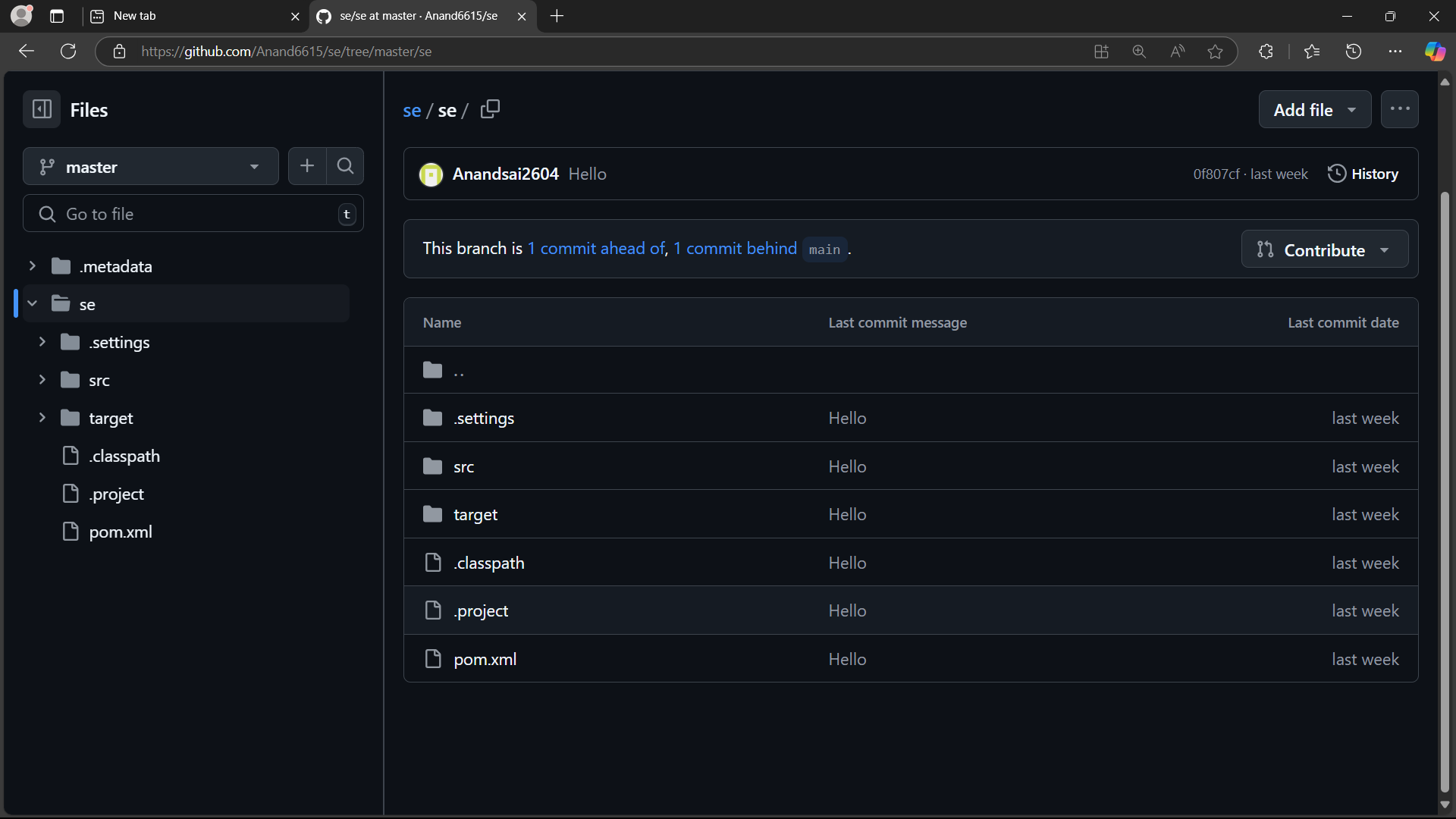


Step 9: Open terminal in Eclipse and run the following commands to initialize a git repository and push the files onto GitHub.





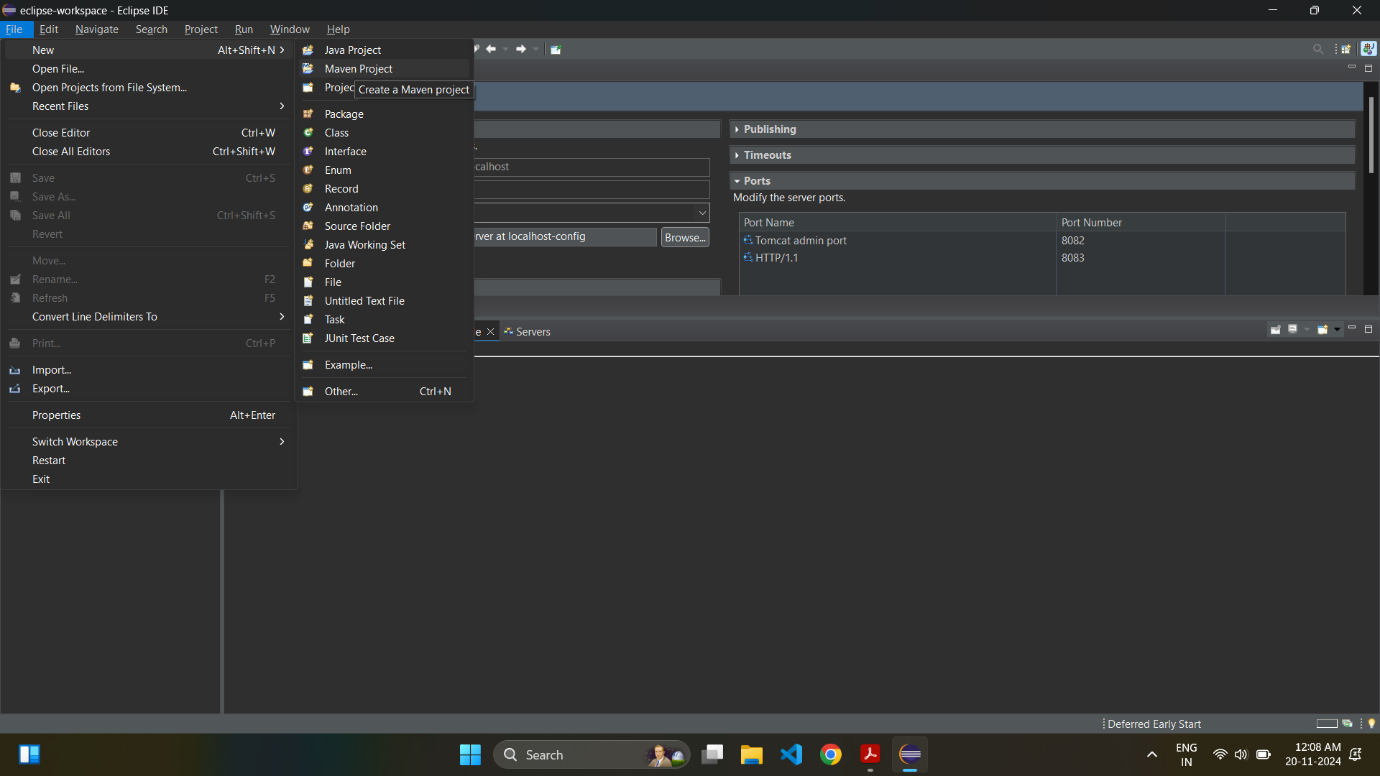
Step 10: The Maven Project files are now pushed onto GitHub.



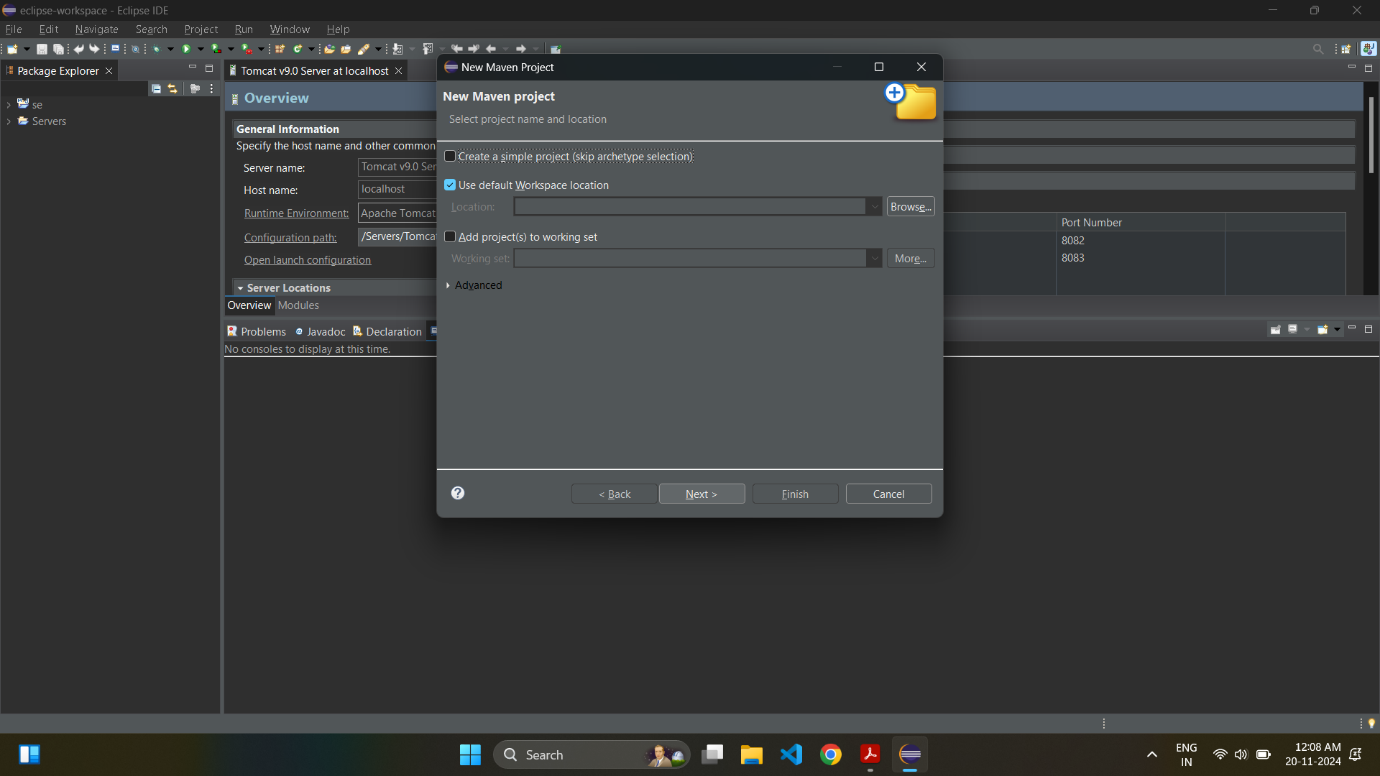
**Exercise 5C: Creating a new Maven Webapp**

* Armaan (22BD1A6611)

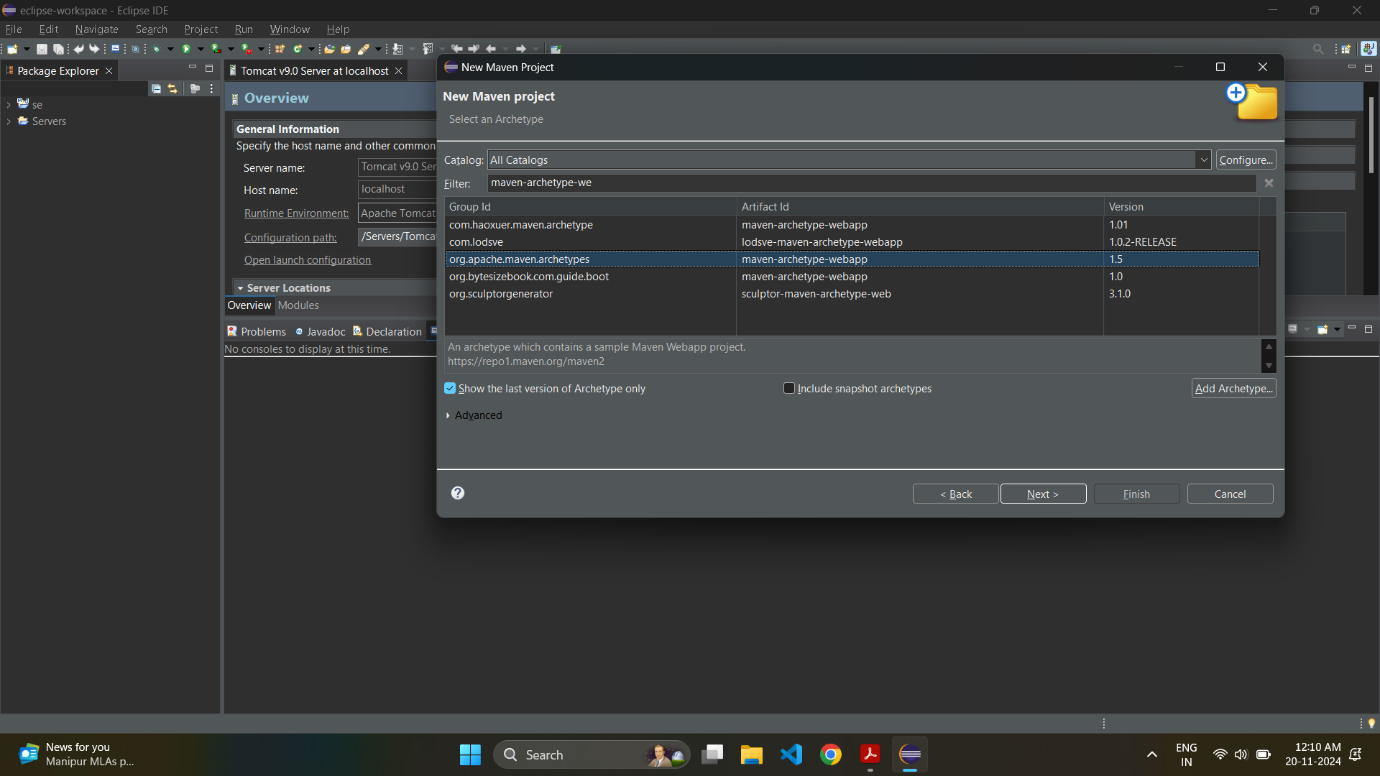
Step 1: In Eclipse, click on File -> New -> Maven Project.



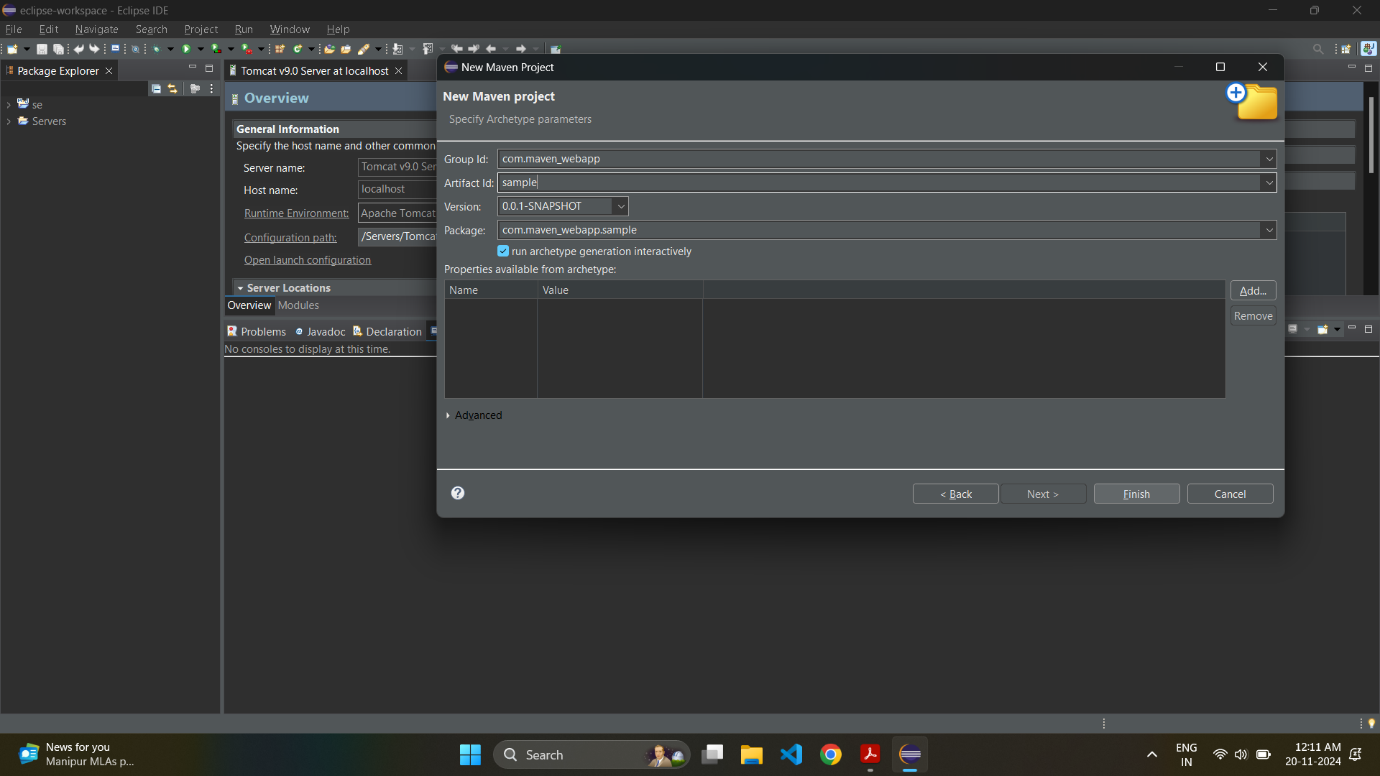
Step 2: Make sure “Use default Workspace location” is checked and click on “Next”.



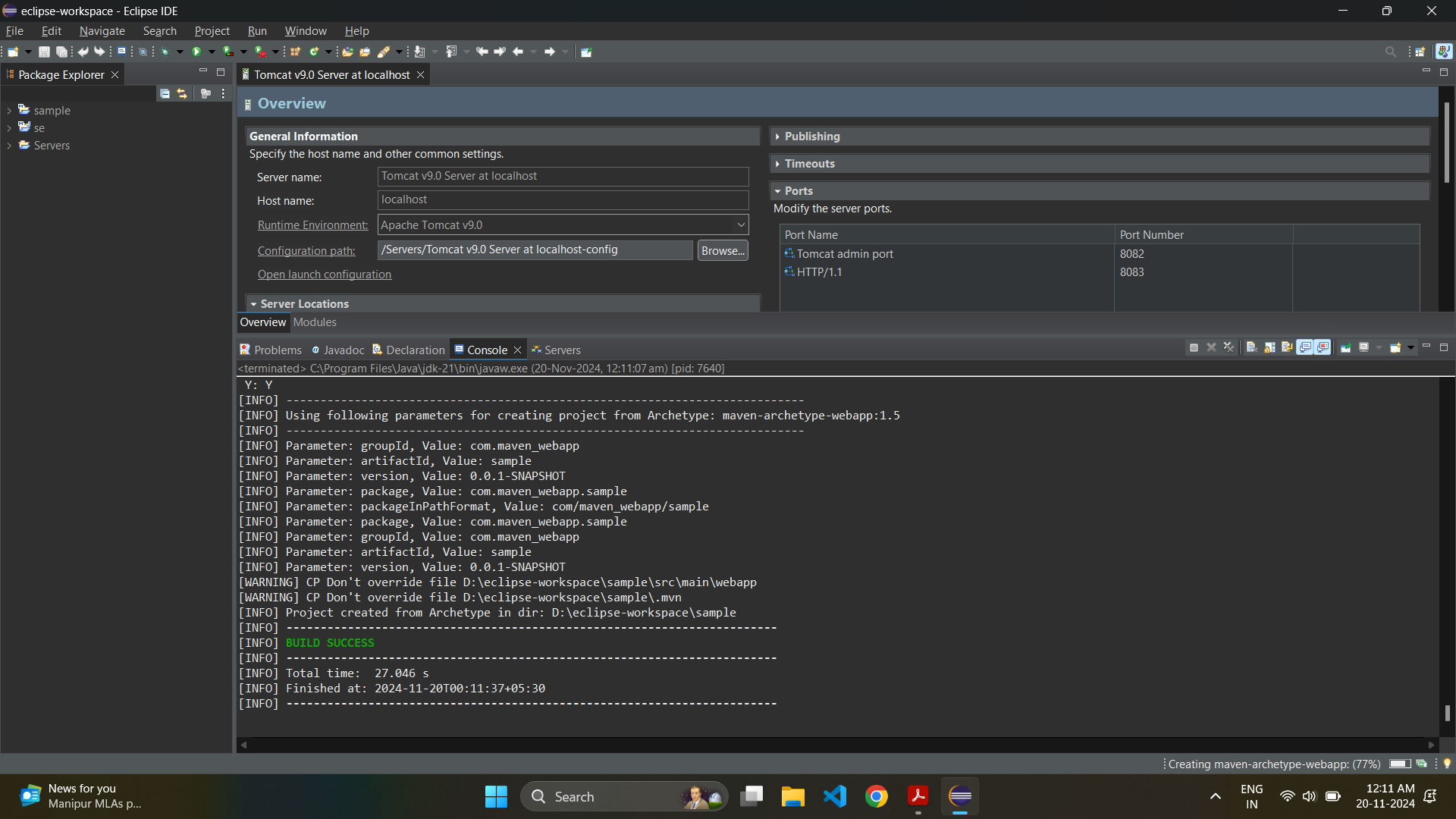
Step 3: When selecting archetype, enter “webapp” in filter and select “maven-archetype-webapp” in the menu below. Click on “Next”.



Step 4: Enter group id in the format com.project\_name and an artifact Id and press on “Finish” to create the project.



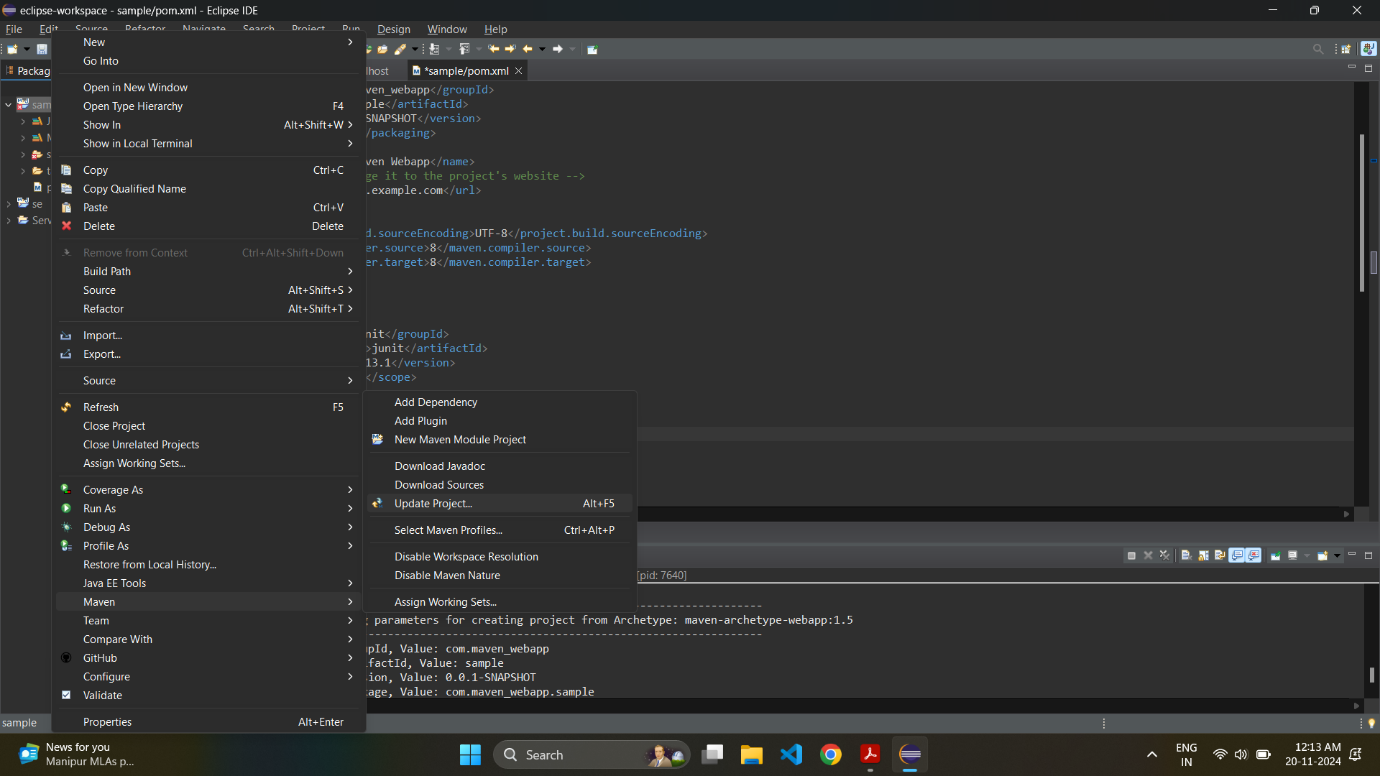
Step 5: You will see “Build Successful” in the console when the build is completed.



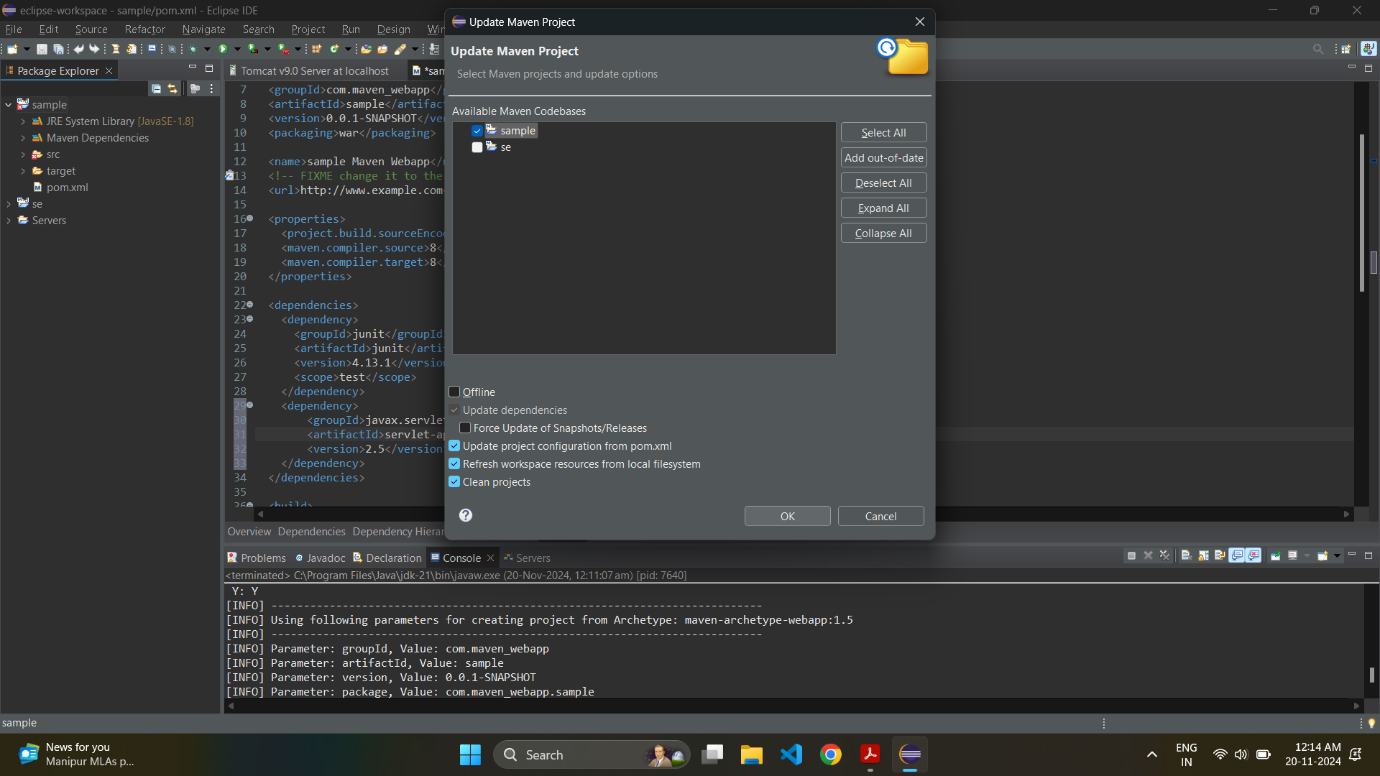
Step 6: After the project is ready, go to project\_name -> pom.xml, and add the highlighted lines of code.



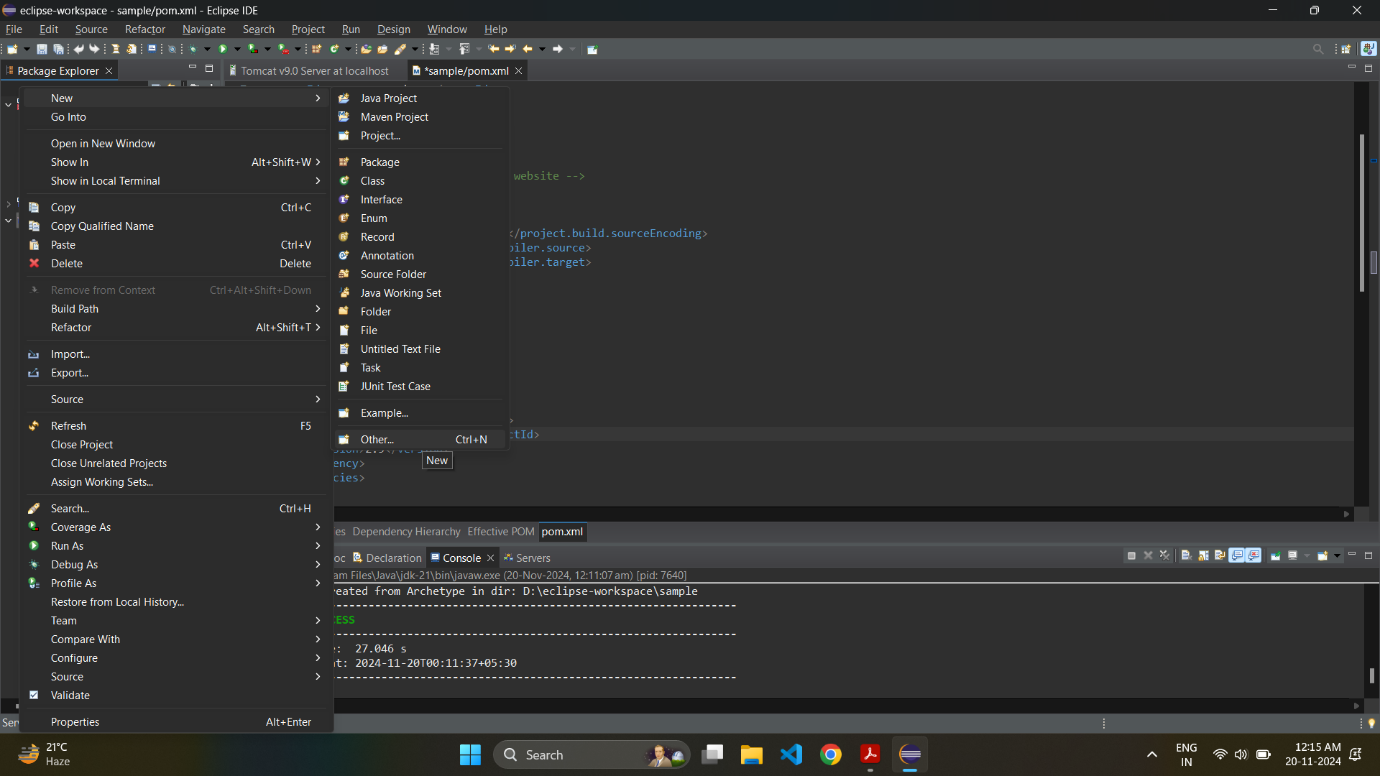
Step 7: Now, right click on the project, go to Maven -> Update Project after saving pom.xml.



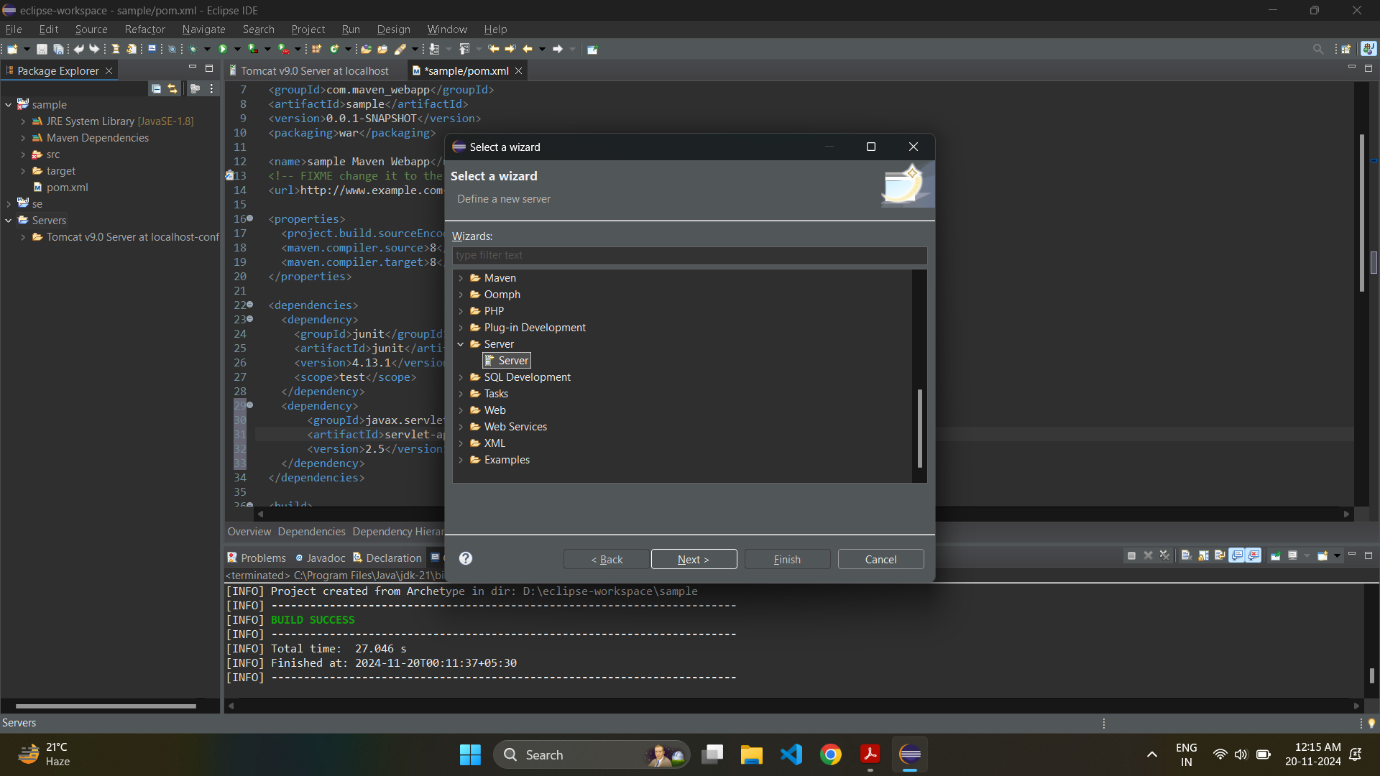
Step 8: Select your project and click on “OK”.



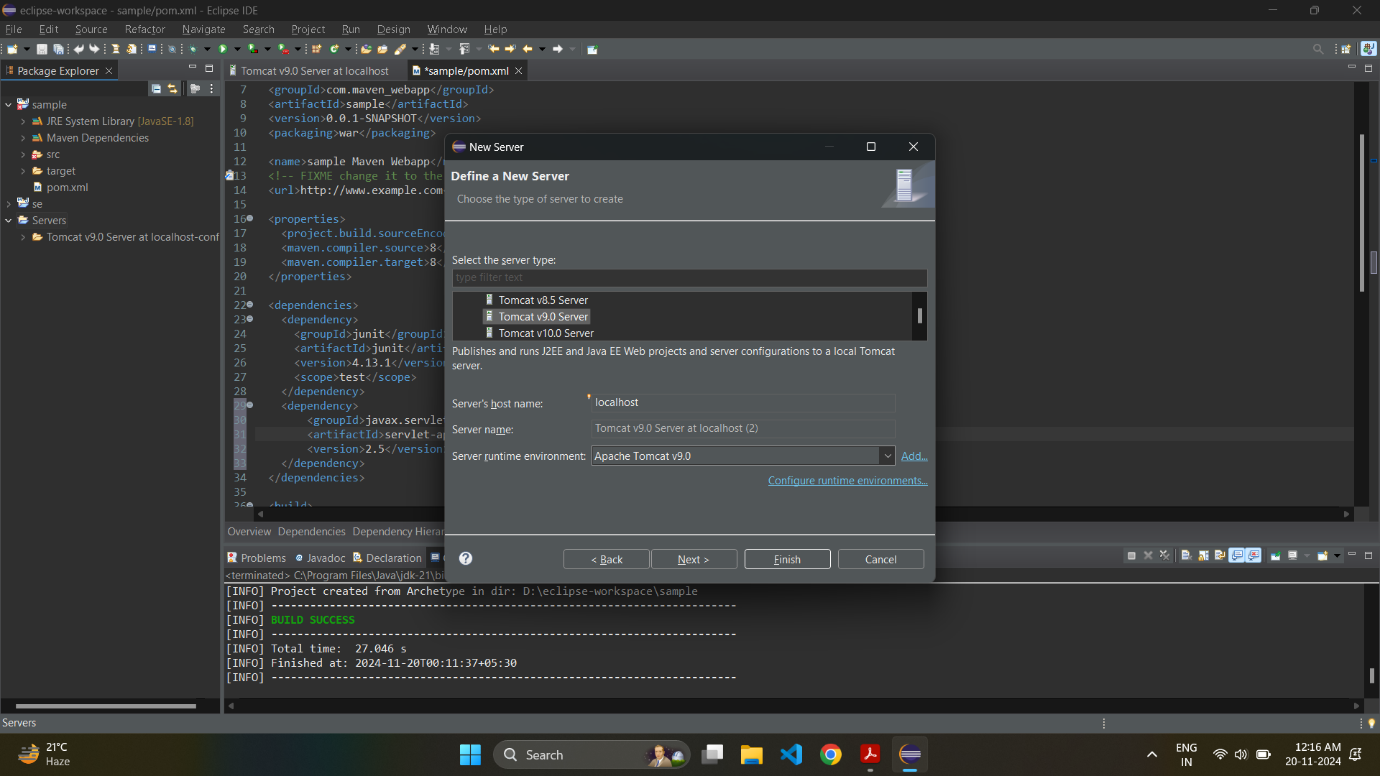
Step 9: Now, create a Tomcat server for your webapp. Right click on the project, New -> Other.



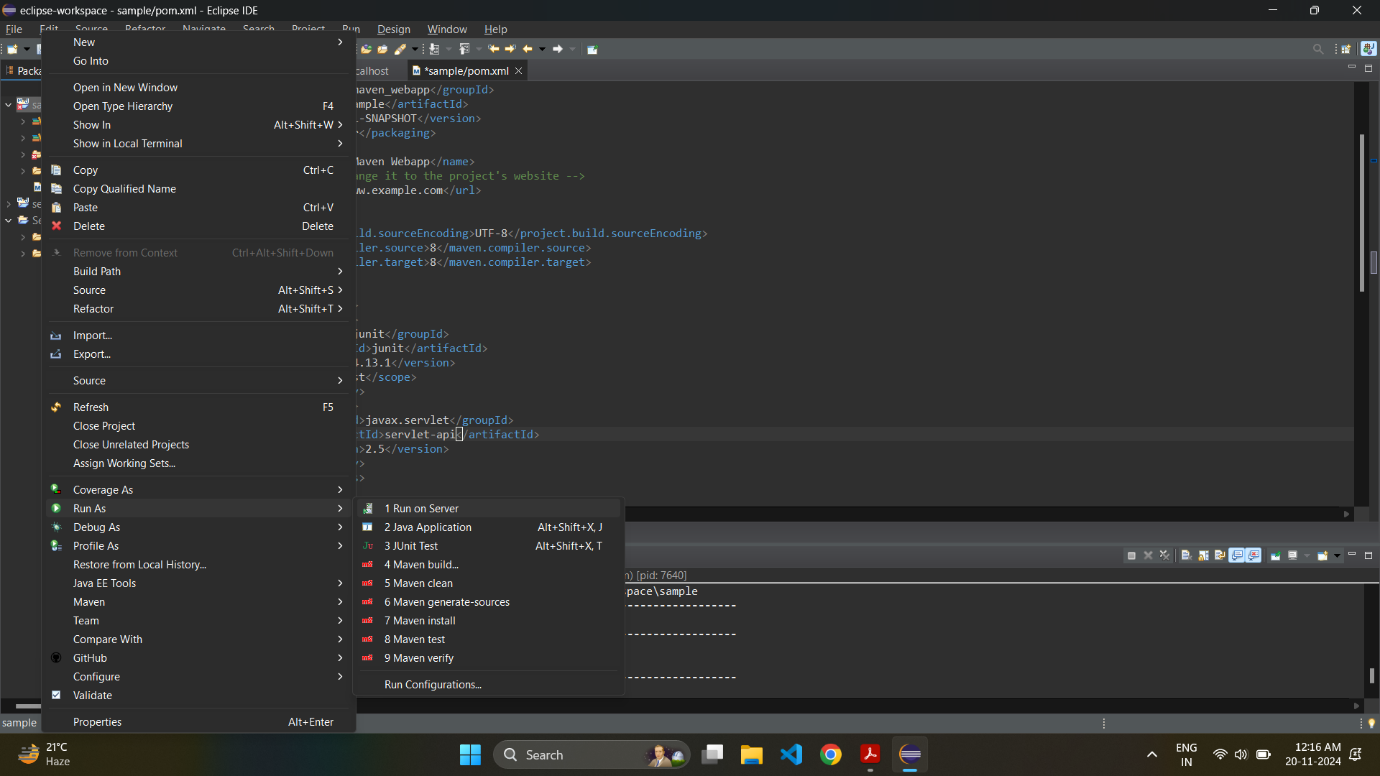
Step 10: Click on Server -> Server.



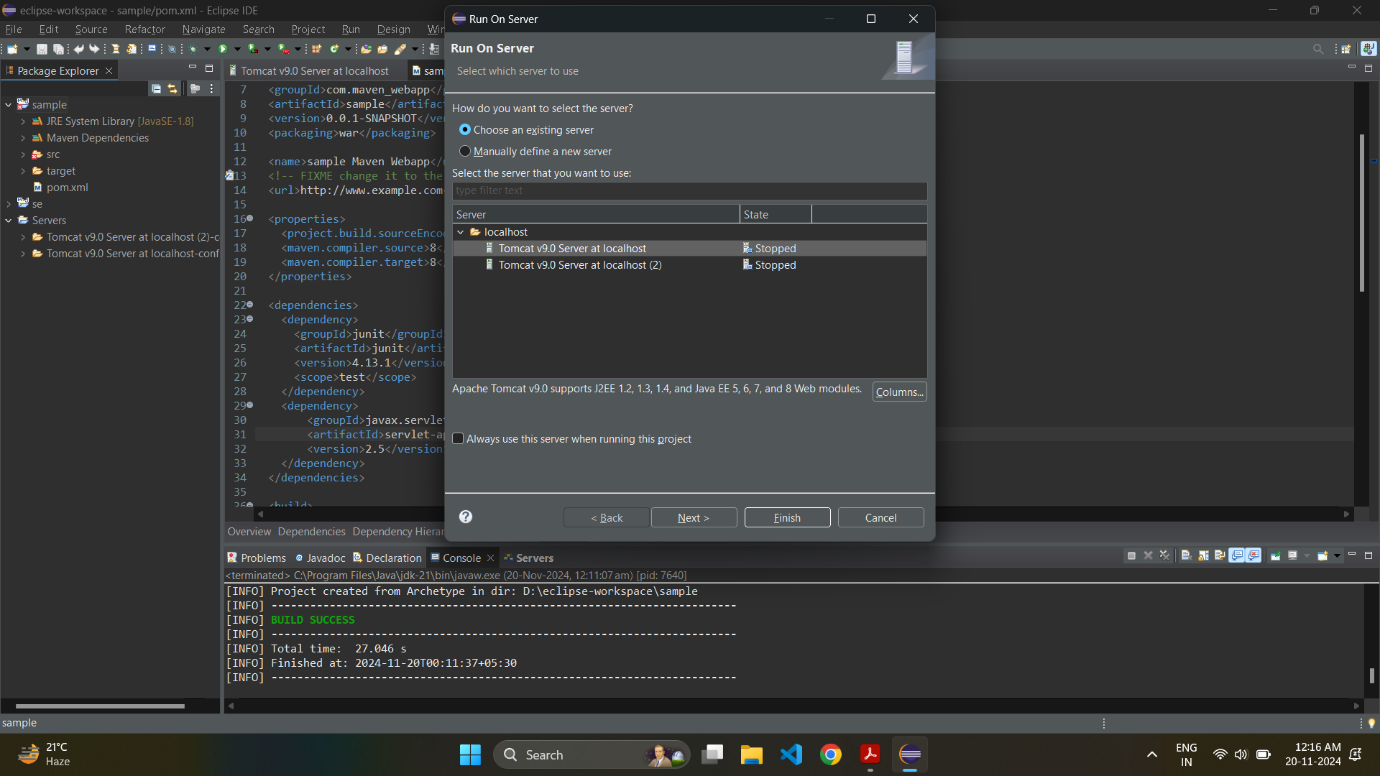
Step 11: Select Tomcat 9.0 Server. Click on Finish.



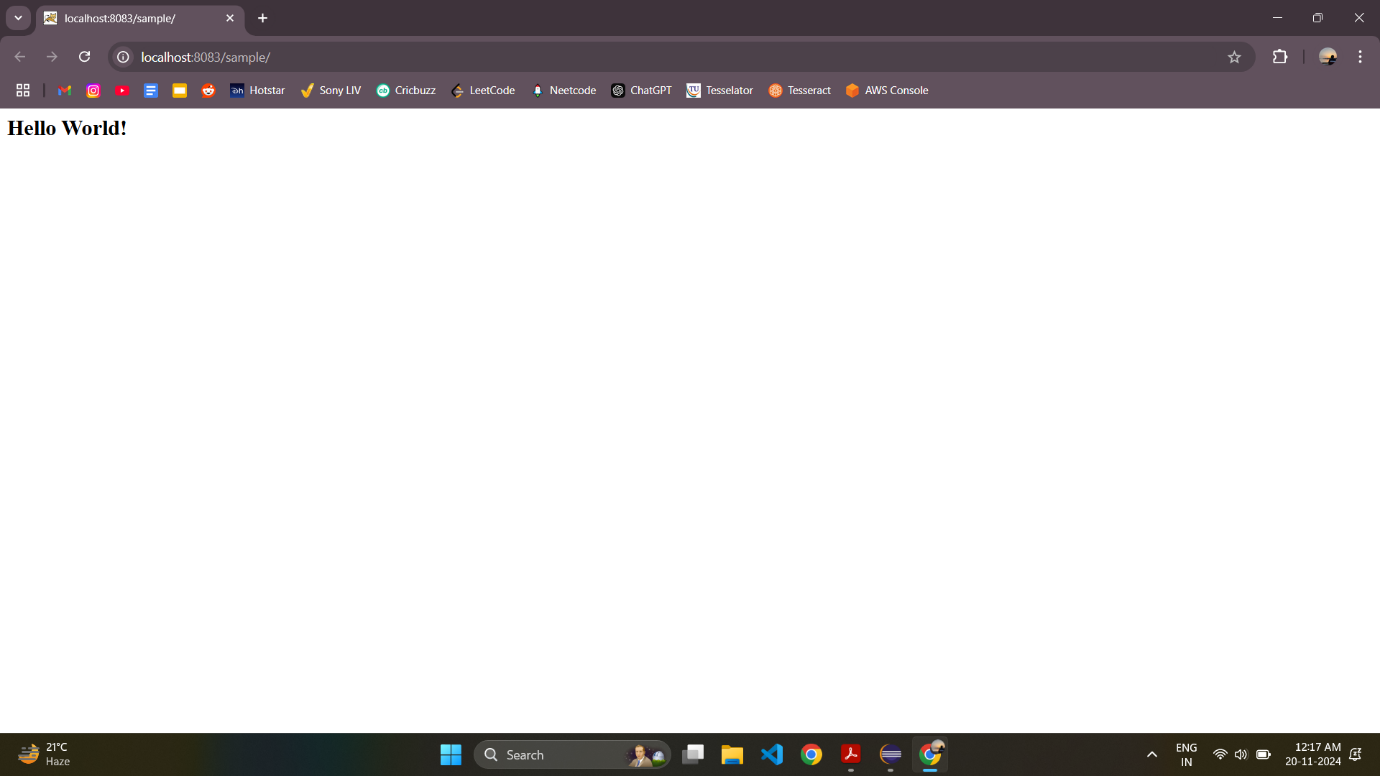
Step 12: To start the application, right click on the project, Run As -> Run on Server.



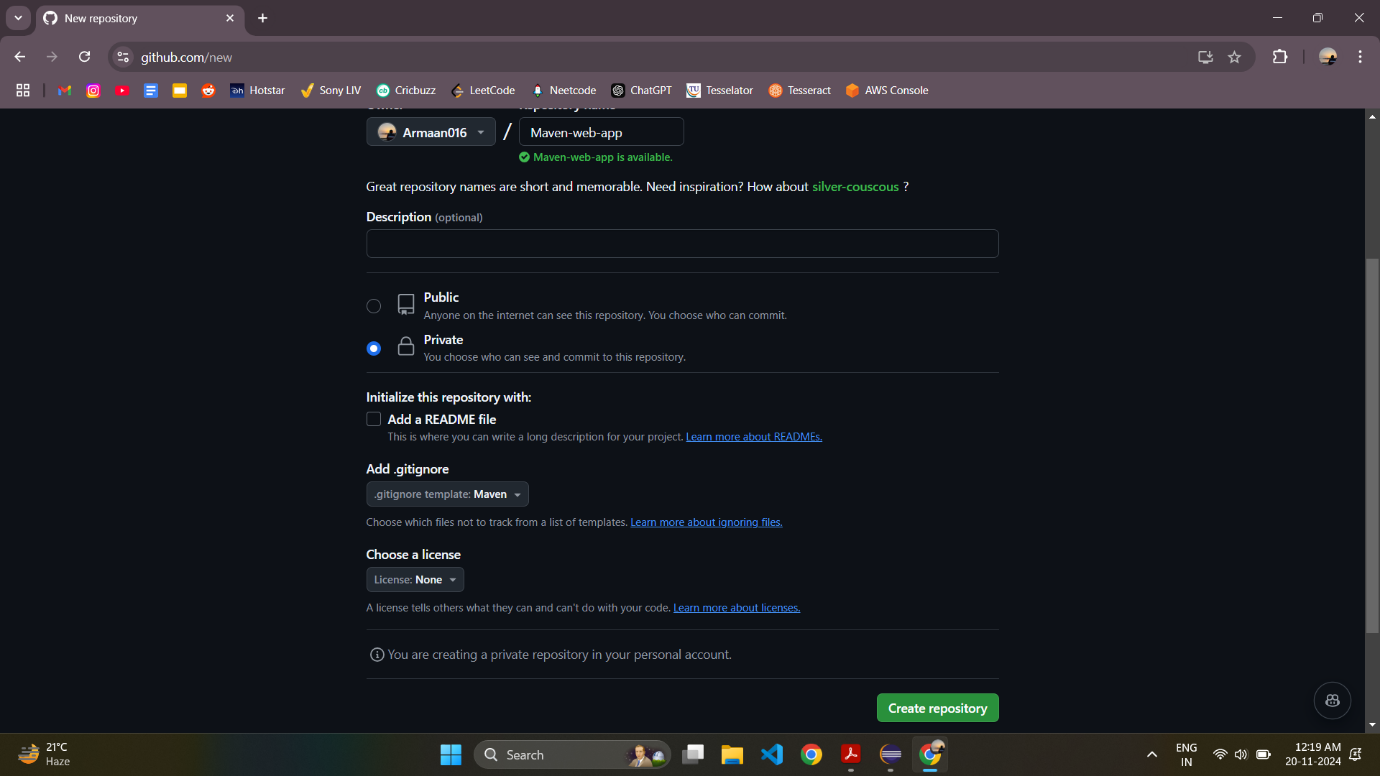
Step 13: Select Tomcat v9.0 Server at localhost, and click on “Finish”.



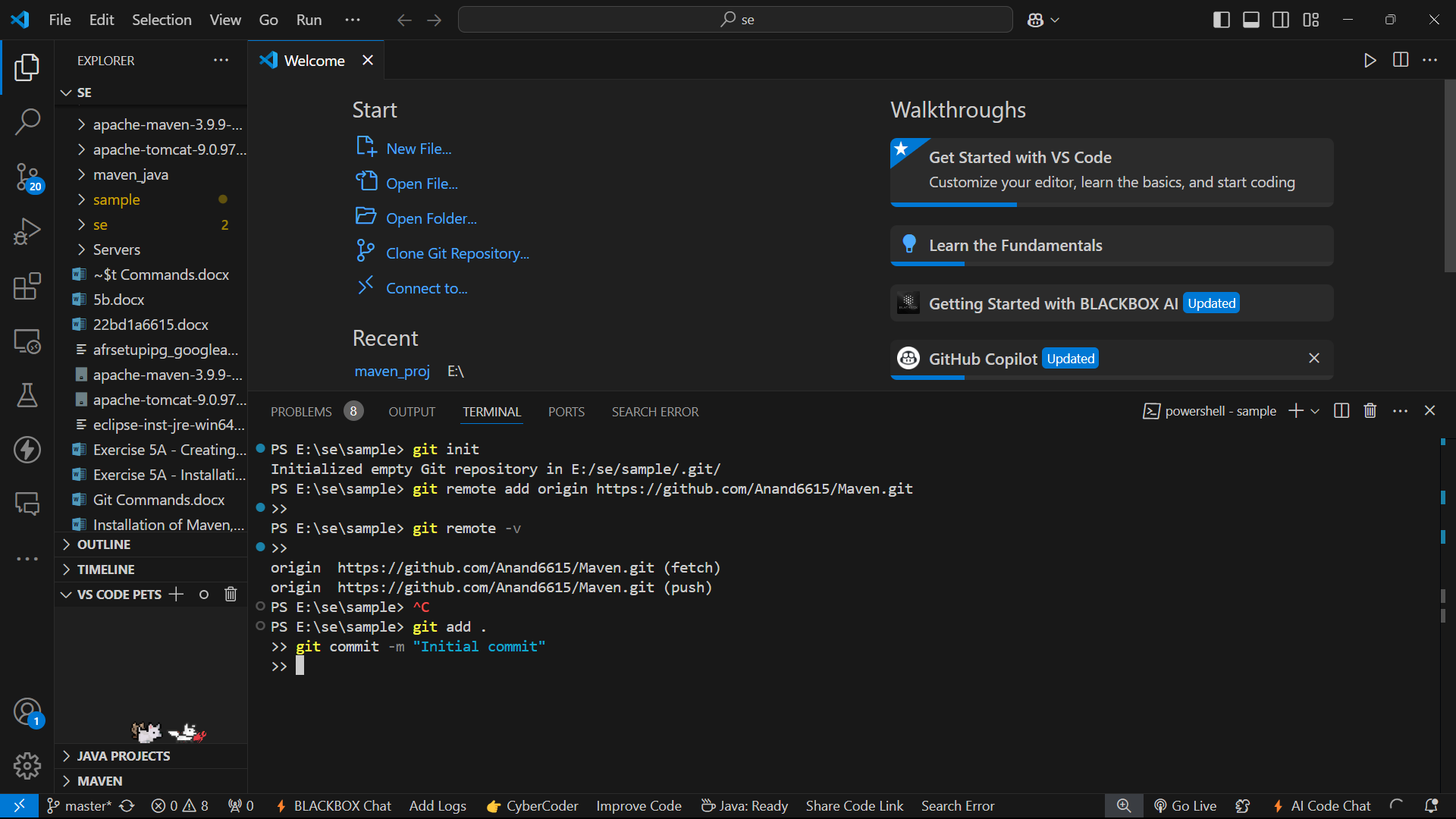
Step 14: The webapp will open and you will see the following screen.



Step 15: To push the code onto GitHub, create a new GitHub repository. Select .gitignore template as Maven, and click on “Create repository”.



Step 16: Navigate to the project directory, initialize a GitHub repository, and push onto GitHub with the following commands.



Step 17: The Maven Project files are now pushed onto GitHub.

