Installation Guide (Masters Project) – Aravind Pagadala

Here is a step-by-step installation guide for setting up Java Maven project in Eclipse IDE on Windows:

Step 1: Install Java Development Kit (JDK)

1. **Download JDK**:
    - Go to the Oracle JDK download page.
    - Download the latest JDK version suitable for Windows.
2. **Install JDK**:
    - Run the downloaded installer.
    - Follow the on-screen instructions to complete the installation.
    - Note the installation directory (e.g., C:\Program Files\Java\jdk-17).
3. **Set Environment Variables**:
    - Open the Control Panel, go to System and Security -> System -> Advanced system settings.
    - Click on Environment Variables.
    - Under System Variables, click New and add JAVA_HOME with the path to the JDK installation directory (e.g., C:\Program Files\Java\jdk-17).
    - Find the Path variable, select it, and click Edit. Add a new entry with %JAVA_HOME%\bin.
4. **Verify Installation**:
    - Open Command Prompt and type java -version and javac -version to ensure Java is installed correctly.

Step 2: Install Eclipse IDE

1. **Download Eclipse**:
    - Go to the Eclipse download page.
    - Download the "Eclipse IDE for Java Developers" package.
2. **Install Eclipse**:
    - Extract the downloaded zip file to a desired location (e.g., C:\Eclipse).
    - Navigate to the Eclipse directory and run eclipse.exe.

Step 3: Install Maven

1. **Download Maven**:
    - Go to the Maven download page.
    - Download the latest binary zip archive.
2. **Install Maven**:
    - Extract the zip archive to a desired location (e.g., C:\Program Files\Maven).
3. **Set Environment Variables**:
    - Open the Control Panel, go to System and Security -> System -> Advanced system settings.
    - Click on Environment Variables.
    - Under System Variables, click New and add MAVEN_HOME with the path to the Maven installation directory (e.g., C:\Program Files\Maven).
    - Find the Path variable, select it, and click Edit. Add a new entry with %MAVEN_HOME%\bin.
4. **Verify Installation**:
    - Open Command Prompt and type mvn -version to ensure Maven is installed correctly.

Step 4: Create a Maven Project in Eclipse

1. **Open Eclipse**:
    - Launch Eclipse by running eclipse.exe (or use start button).
2. **Install Maven Integration for Eclipse**:
    - Go to Help -> Eclipse Marketplace.
    - In the Eclipse Marketplace dialog, search for "Maven Integration for Eclipse" (also known as m2e).
    - Click Go and install the plugin.
3. **Create a New Maven Project**:
    - We might not need to create a Maven project, instead we open an already existing Spring Boot project.
    - Go to File -> Import.
    - In Import, select ‘Existing Maven Project’, in the next page under Root directory -> choose Browse and open the project which contains pom.xml file.

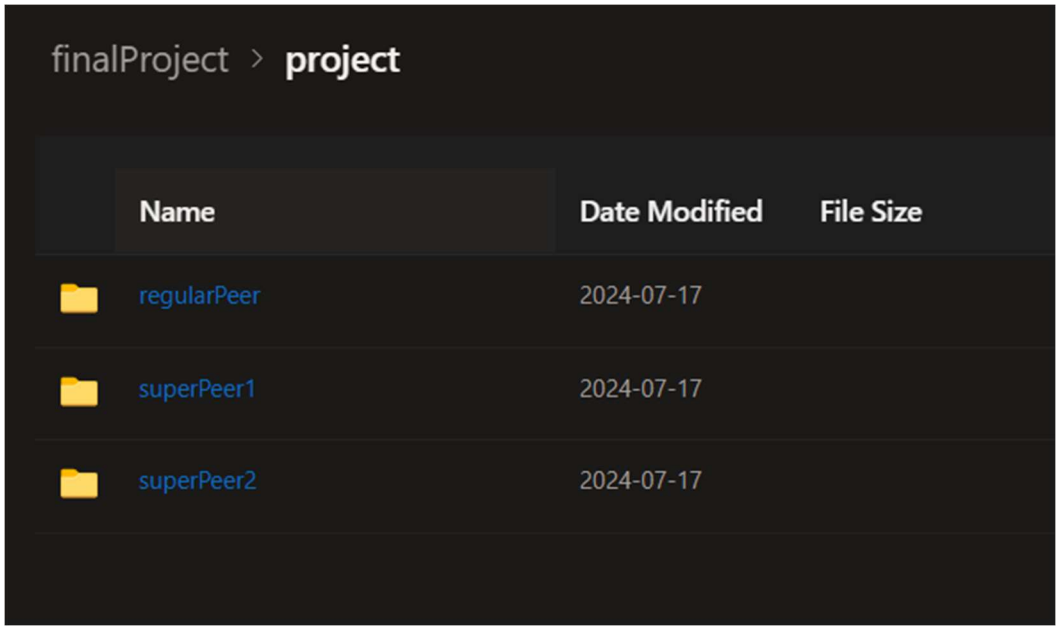
Steps to Install the project and Project Setup:

Please follow the below steps to set up the project.

I assume the above steps led you to install all the software tools to configure this project. As we have installed Java, Maven and Eclipse IDE, let's continue to setup.

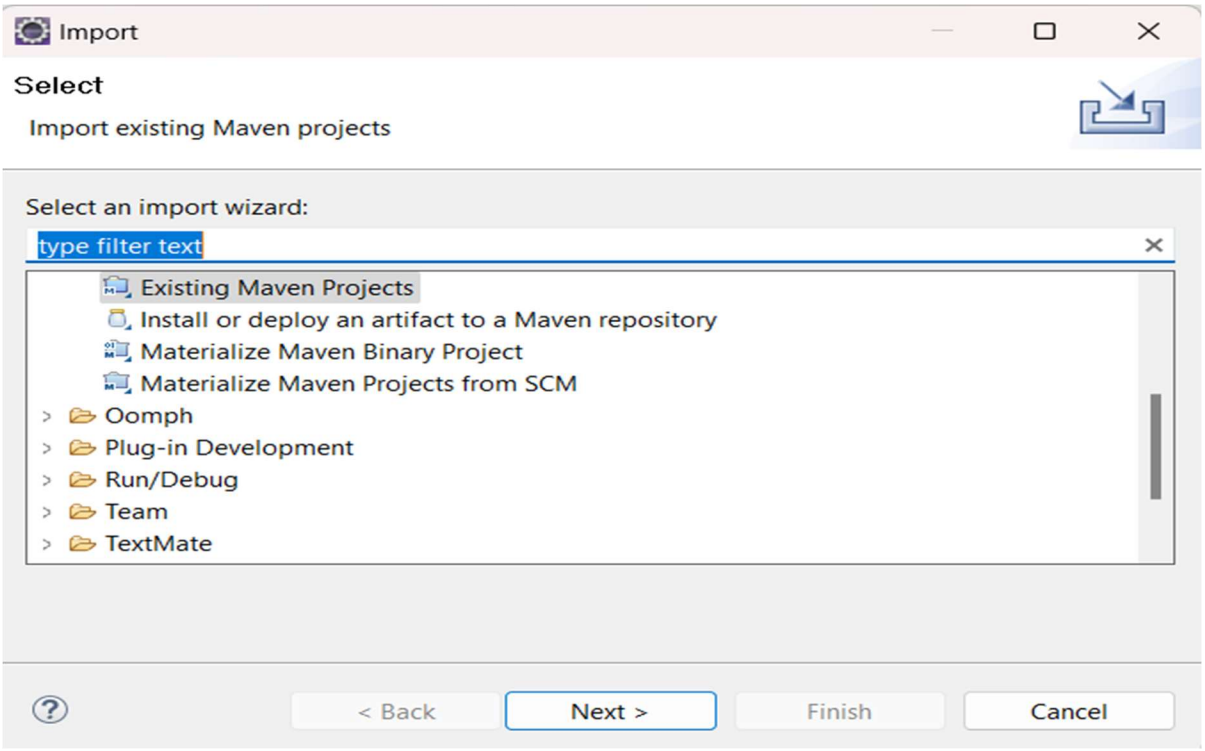
Step1: Download the Source code

- Please download the source code for this project, I have sent the code in a .zip file.
- If it is not available, please find the link for source code here.
- Extract the project files. (zip file should have three folders, extract them all).

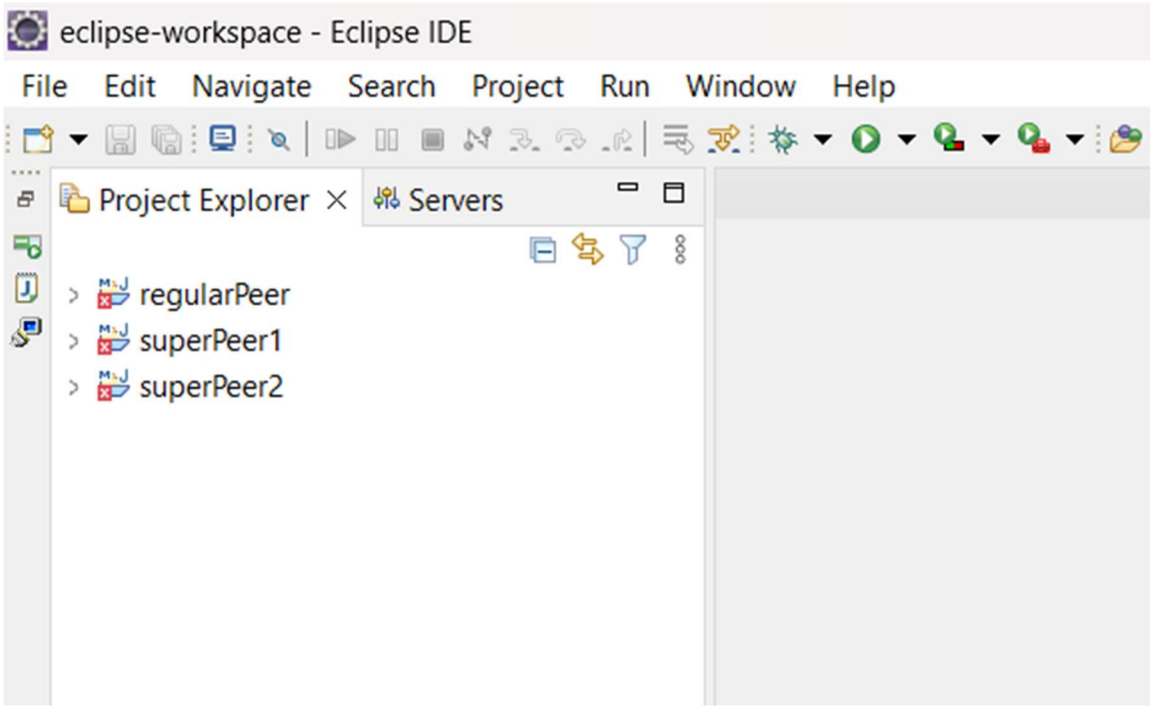


Step 2: Importing Projects

- Open the Eclipse IDE, go to File → 'import'.
- Choose 'Existing Maven Project', click next.

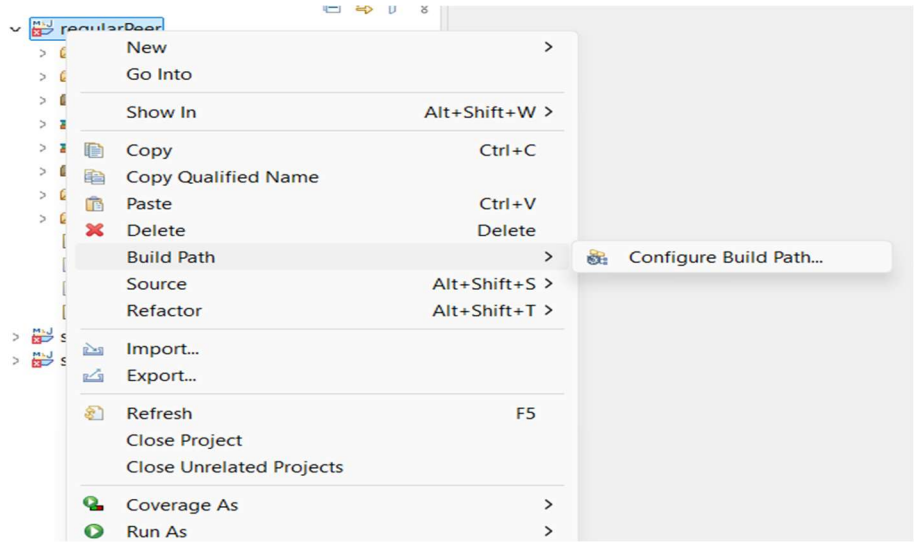


- Import all the three projects present in the extracted folder of source code, Click Finish.
- By now the Eclipse IDE should look like this. (Contains Maven Projects regularPeer, superPeer1, superPeer2)

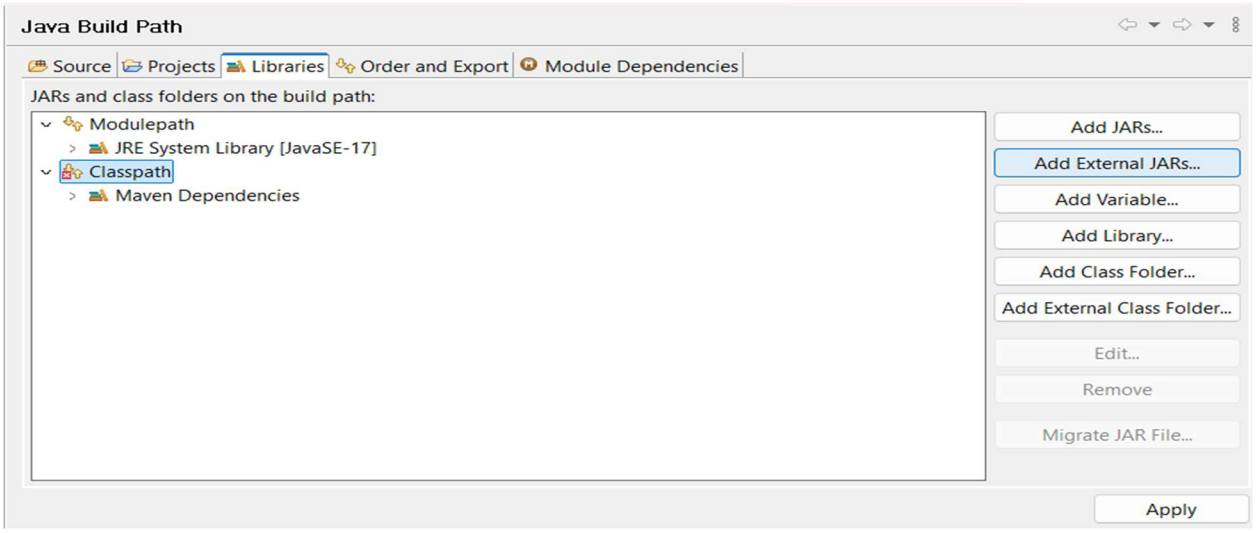


Step 3: Add dependencies

- Download the dependencies file that is sent in email, or you can also find the file in source code .zip. It contains commonInterface.jar file only.
- Note: This jar file contains only the common Interface, no Implementation code included.
- Extract the dependencies zip file to local PC.
- As of now you can see errors in the projects due to missing dependency file, we will include the jar file into each other projects.
- Right-click on regular peer project, go to build path → configure Build Path.



- Under libraries, click Classpath, Click add External JARs on the right menu.

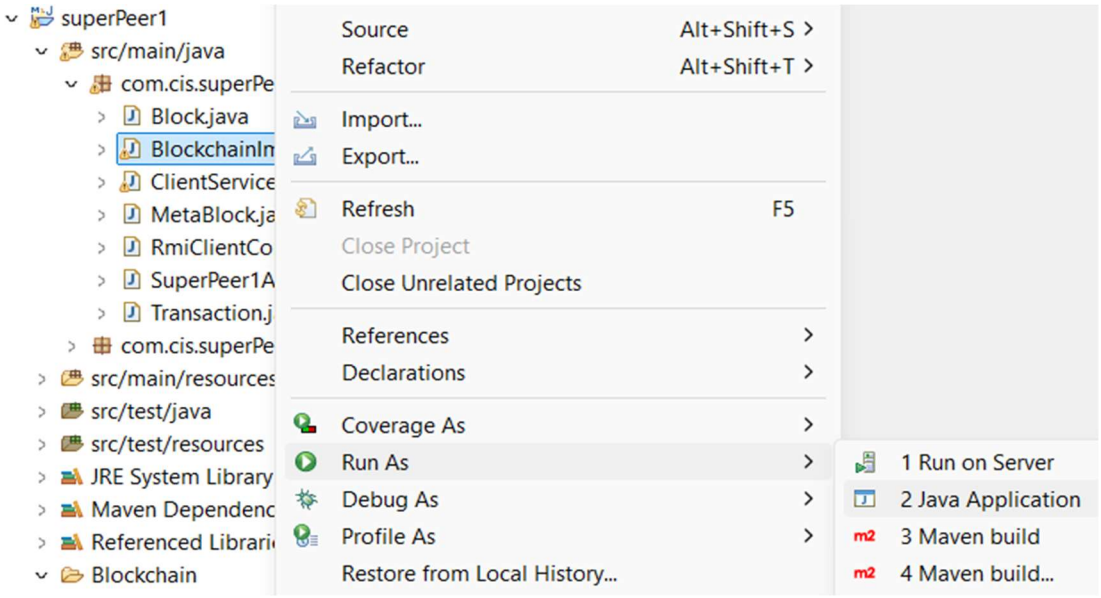


- Do same for superPeer1and superPeer2, include commonInterface.jar file in classpath.

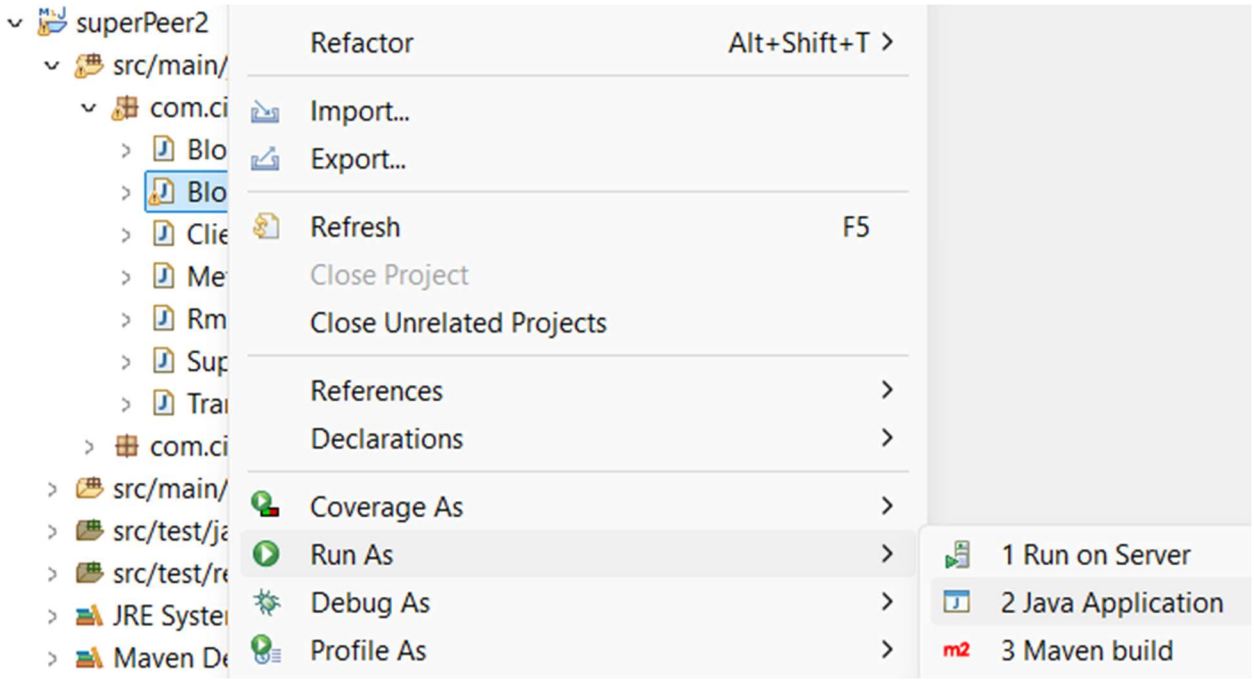
Running the Application:

Run the Server Applications:

1. **Run com.cis.superPeer1/BlockchainImpl**:
    - In the superPeer1 Project, open the com.cis.superPeer1 package
    - Right-click on the class BlockchainImpl.
    - Select Run As > Click Java Application.



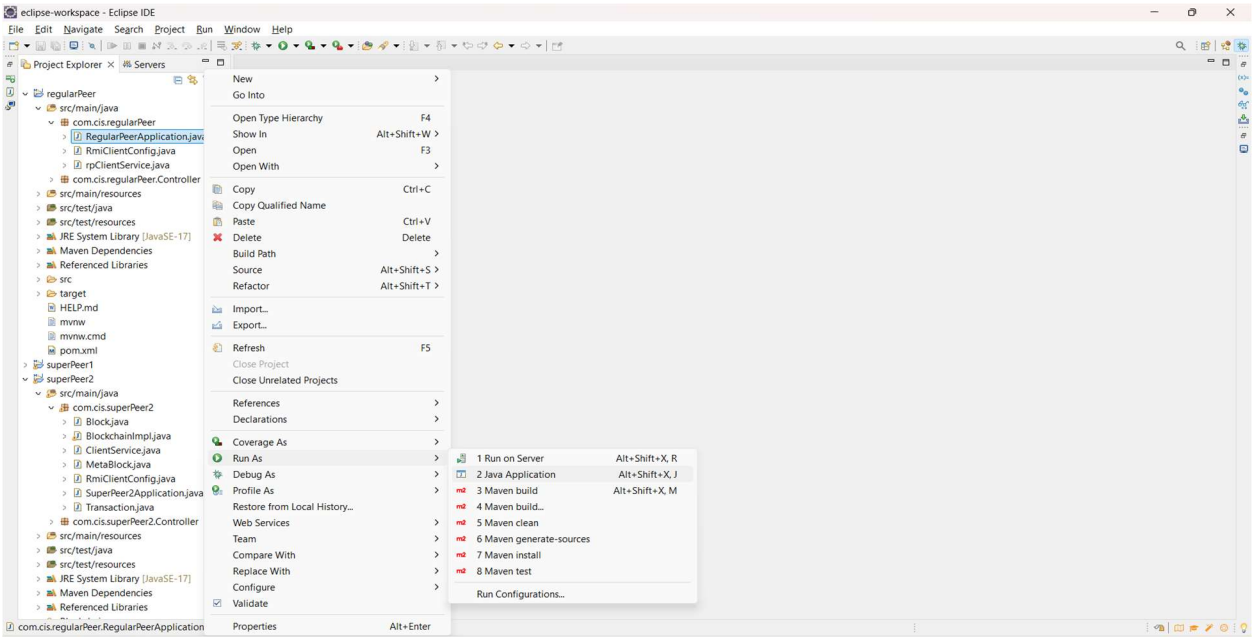
2. **Run com.cis.superPeer2/BlockchainImpl**:
    - In the superPeer2 Project, open the com.cis.superPeer2 package
    - Right-click on the class BlockchainImpl.
    - Select Run As > Click Java Application.



Run Client Configurations:

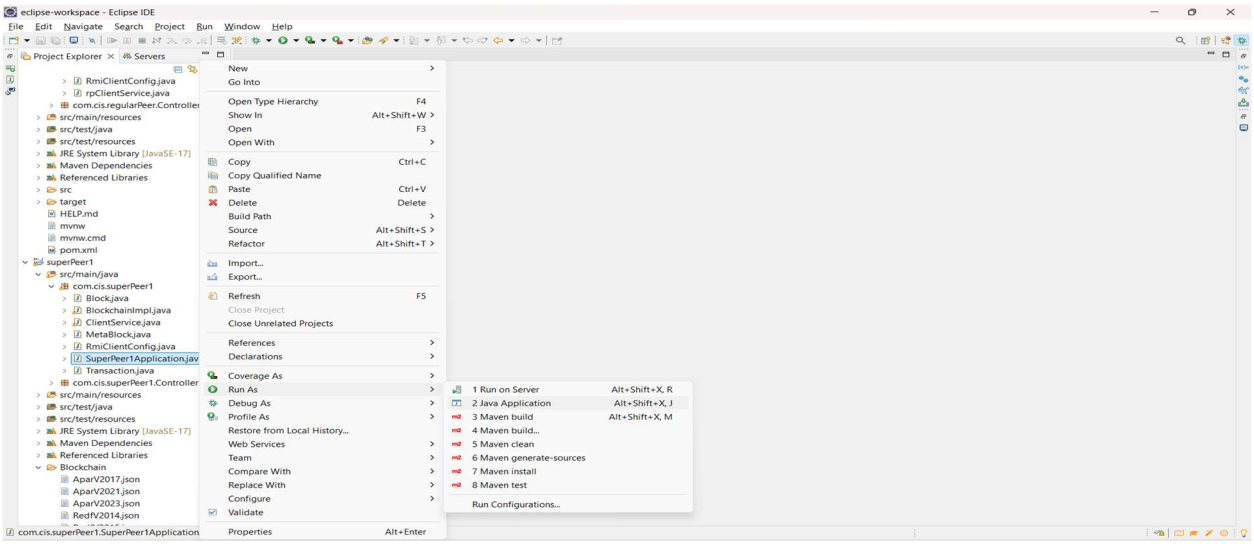
1. Run the Regular Peer Application:

- In the regularPeer Project, open the com.cis.regularPeer package
- Right-click on the RegularPeerApplication.java
- Select Run As > Click Java Application.



2. **Run the Super Peer 1 Application:**

- In the superPeer1 Project, open the com.cis.superPeer1 package
- Right-click on the SuperPeer1Application.java
- Select Run As > Click Java Application.



3. **Run the Super Peer 2 Application:**

- In the superPeer2 Project, open the com.cis.superPeer2 package
- Right-click on the SuperPeer2Application.java
- Select Run As > Click Java Application.