1. JDK configuration - Environment variables setup  
   Once the JDK is installed in system, configure the Environment variables.

* Add the JAVA\_HOME variable as:

**JAVA\_HOME = C:\Program Files\Java\jdk1.8.0\_112**

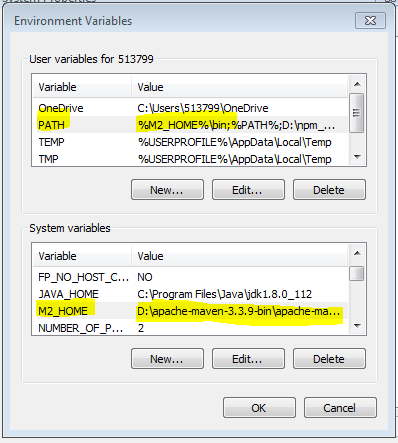
* Append the JAVA\_HOME variable in the Path variables, as:

**Path = %JAVA\_HOME%\bin;**

* To check Go to your command line and type the following command:

**javac -version**

1. Maven configuration.  
   Once the maven is installed in system, configure the Environment variables.

* Add the M2\_HOME variable as:  
  **M2\_HOME = D:\apache-maven-3.3.9-bin\apache-maven-3.3.9**
* Append the M2\_HOME in the PATH variable as, **%M2\_HOME%\bin;**  
  

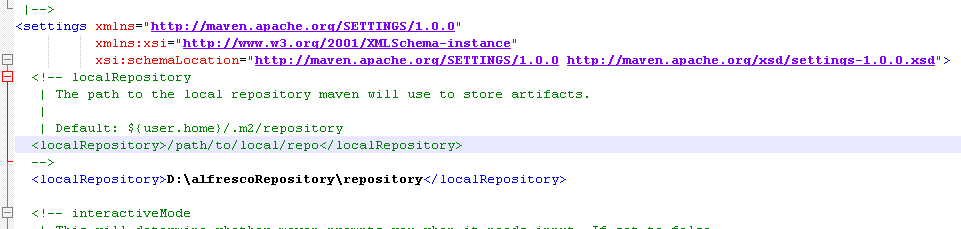
1. Eclipse download and install.
2. Copy "repository" and "demo-repo", "demo-share" folders from the **\\10.242.250.33\training** path.

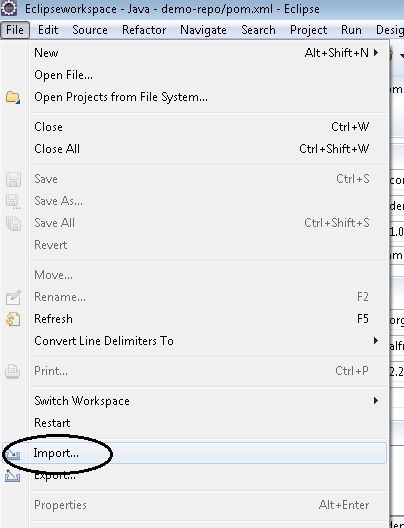
[\\10.242.250.33\training\dependencies\repository](file:///\\10.242.250.33\training\dependencies\repository)

[\\10.242.250.33\training\projects\demo-repo](file:///\\10.242.250.33\training\projects\demo-repo)

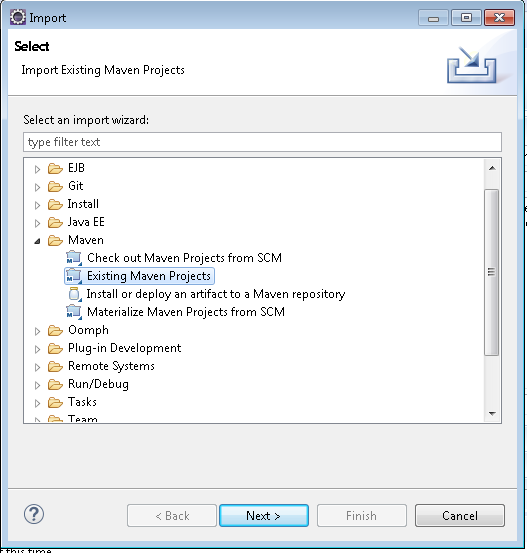
[\\10.242.250.33\training\projects\demo-share](file:///\\10.242.250.33\training\projects\demo-share)

1. Extract and place the "**repository**" in a specific path.   
   Suppose, **D:\alfrescoRepository\repository**
2. Open the settings.xml file : D:\apache-maven-3.3.9-bin\apache-maven-3.3.9\conf\settings.xml and append the localRepository path :  
    **<localRepository>D:\alfrescoRepository\repository</localRepository>**

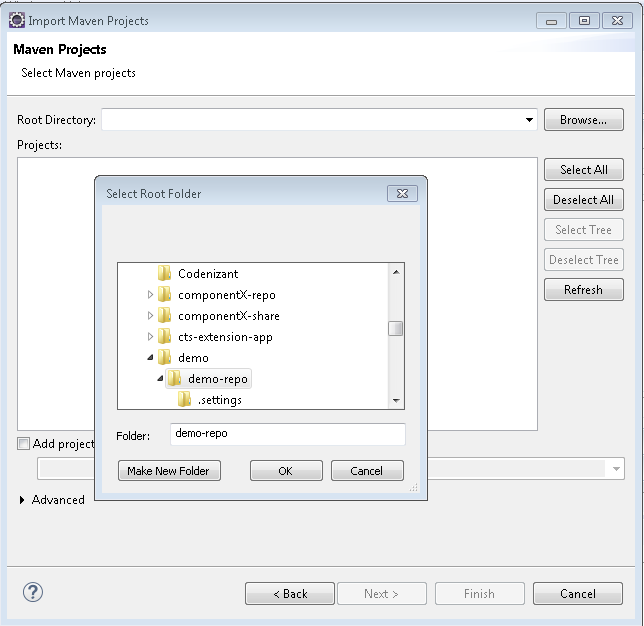


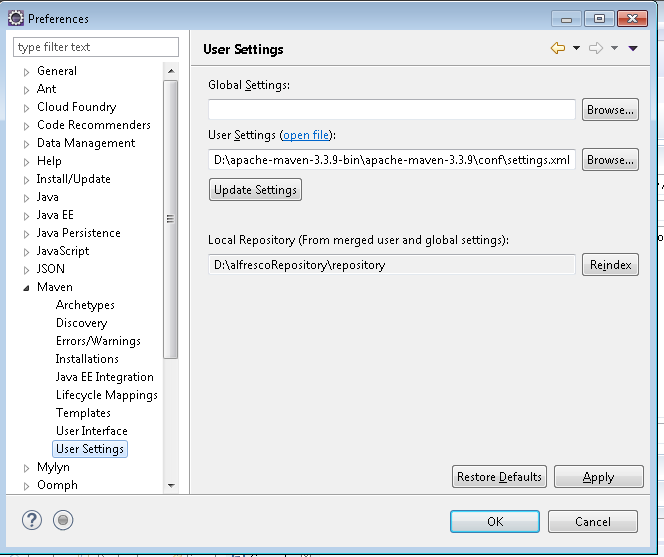
1. Create a folder, named "demo" in D: and extract and place the "demo-repo" and "demo-share" inside that path.
2. Open Eclipse, import the project as Maven Project. Select **demo-repo** and **demo-share** projects. Imported the projects in your eclipse.  
   

Select the Maven >> Existing Maven Project:

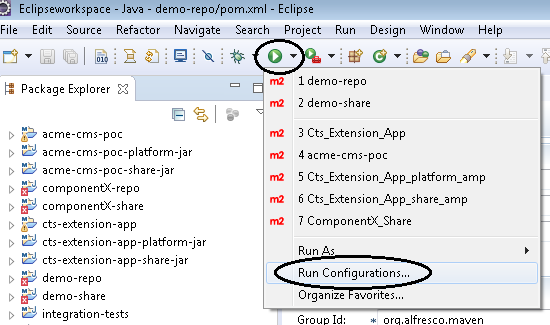
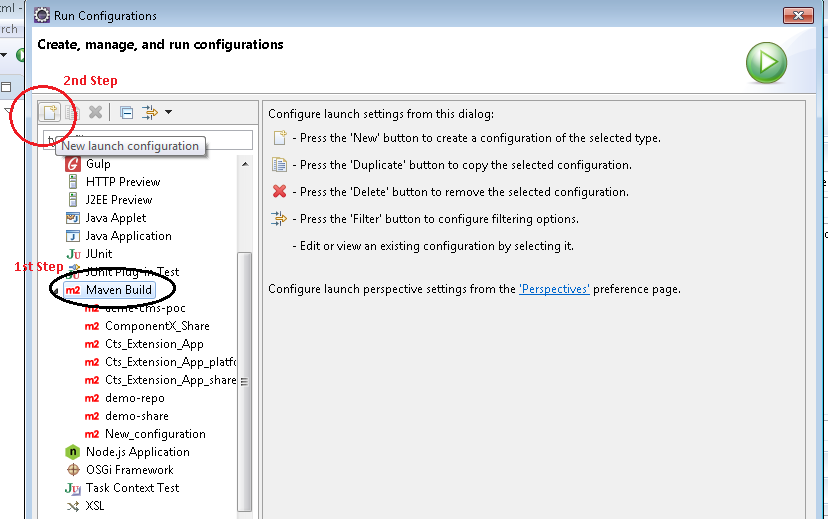
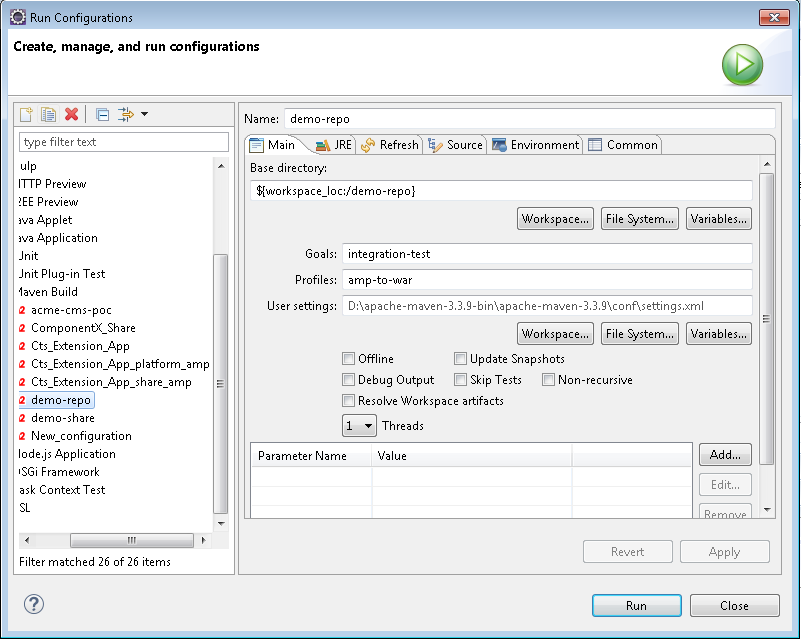


1. Click **Next** and **Browse** to the particular “**demo-repo**” and “**demo-share**” project as:

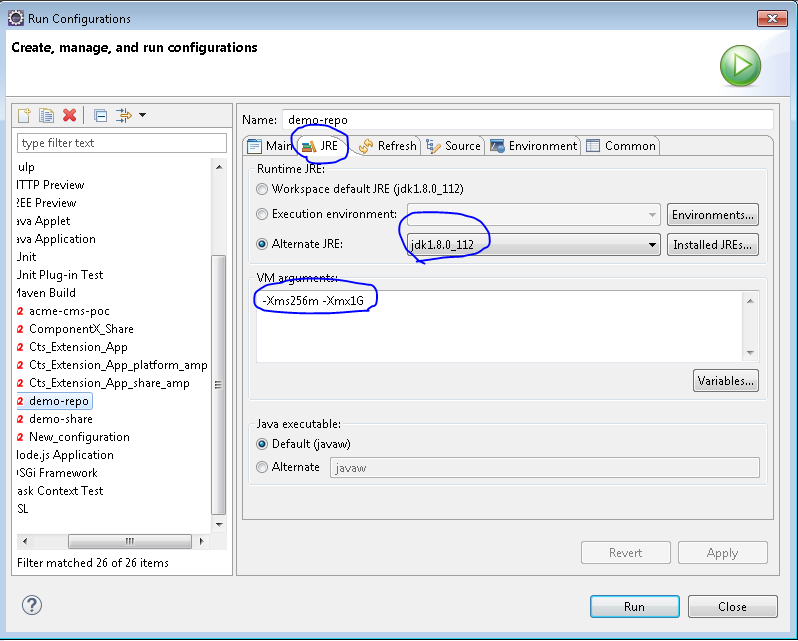


1. Once importing is done for both the projects, we need to compile the projects. Open Command Prompt >> Navigate to the demo-repo project >> run “**mvn clean install**”:   
     
   Similarly we need to do clean install for the **demo-share** project.
2. Once the build is successful for both the projects, open Eclipse and navigate to **Window >> Preferences >> Maven >> User Settings :**

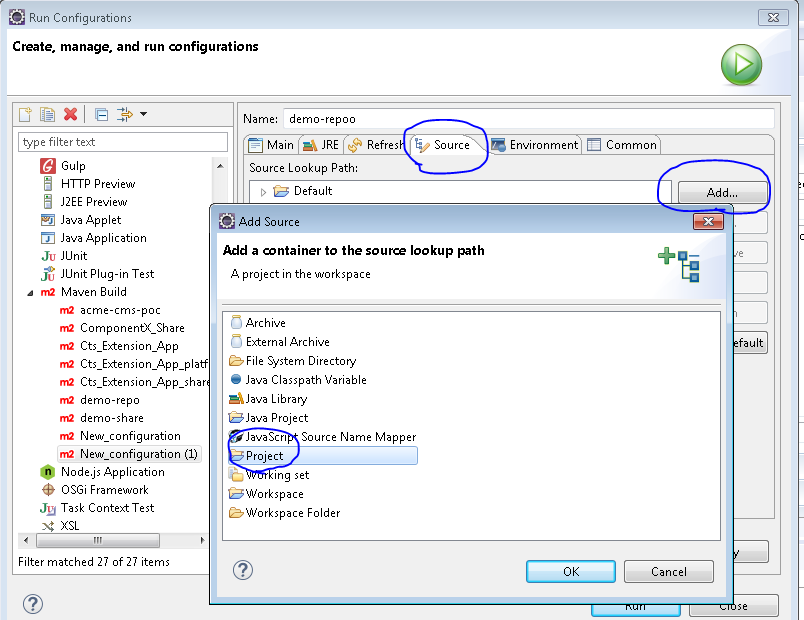
Make sure the settings.xml file is referring to the apache maven folder’s settings.xml file. If not, browse to the particular folder path, and select the settings.xml file.

1. Once the setting is done, next step is to run the alfresco repo and share projects. In Eclipse, click on “**Run Configurations**” :  
   
2. Click on **Maven Build >> New Launch Configuration** as shown below :  
   
3. The below popup will appear, where we need to mention the following values:  
    Name : demo-repo  
    Base directory : Click on the Workspace and browse the “demo-repo” path  
    Goals : integration-test  
    Profiles : amp-to-war  
   

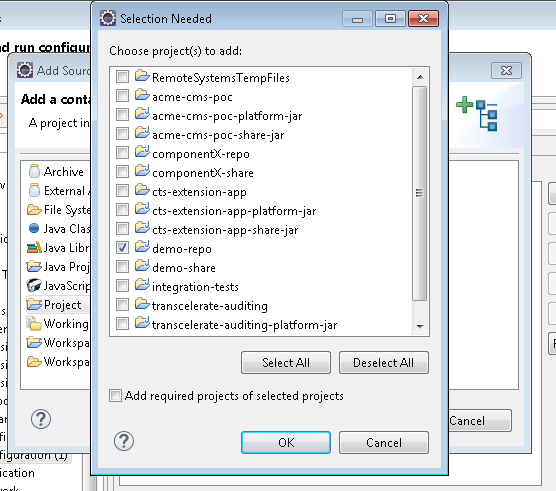
Next in the JRE tab, in the same popup, check whether JDK proper version is installed or not.



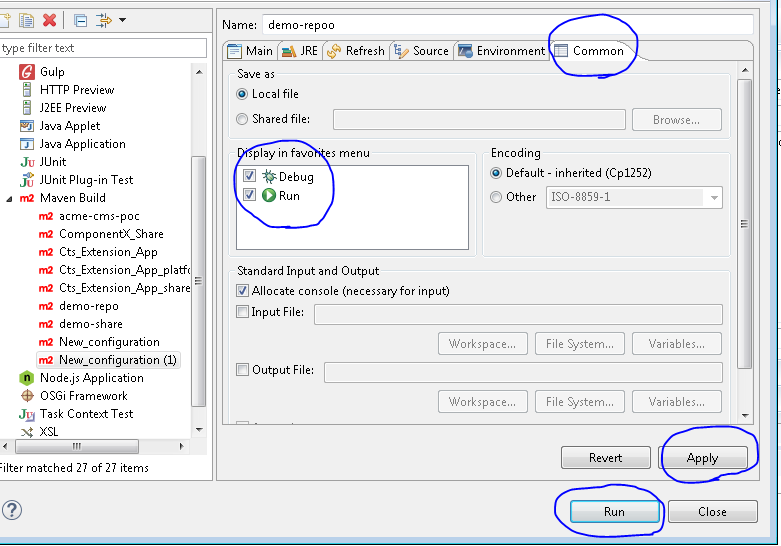
Next in the **Source** tab, only **Default** project will be there, we need to select the **demo-repo** project as well:



Once click on “OK”, a new selection window will appear, where we need to select the **demo-repo** project:



Next click on the **Common** tab, check the Debug and Run from the Favorites Menu >> Apply and Run:



The same procedures from step no.12 to step no.14 should be followed for running the **demo-share** project.

Once it is up and running, we can login to our localhost,

<http://localhost:8082/share>

Provide the admin credentials, by default,   
**user name: admin  
password: admin**