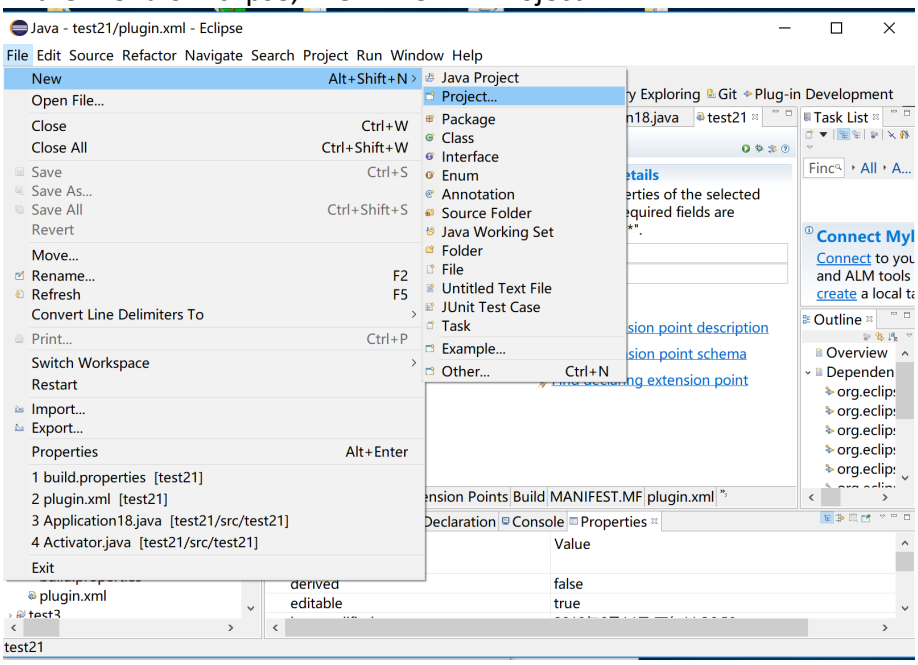
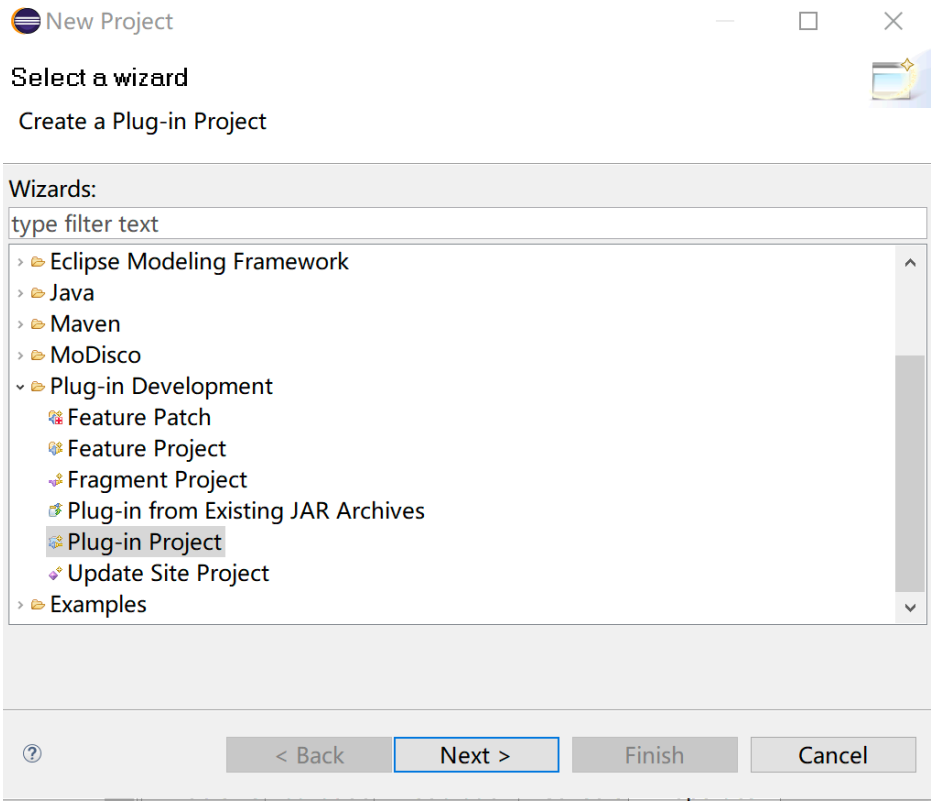
Tutorial To Create the Headless Plugin “Extractor” In Eclipse

## 1. Create a new plugin project

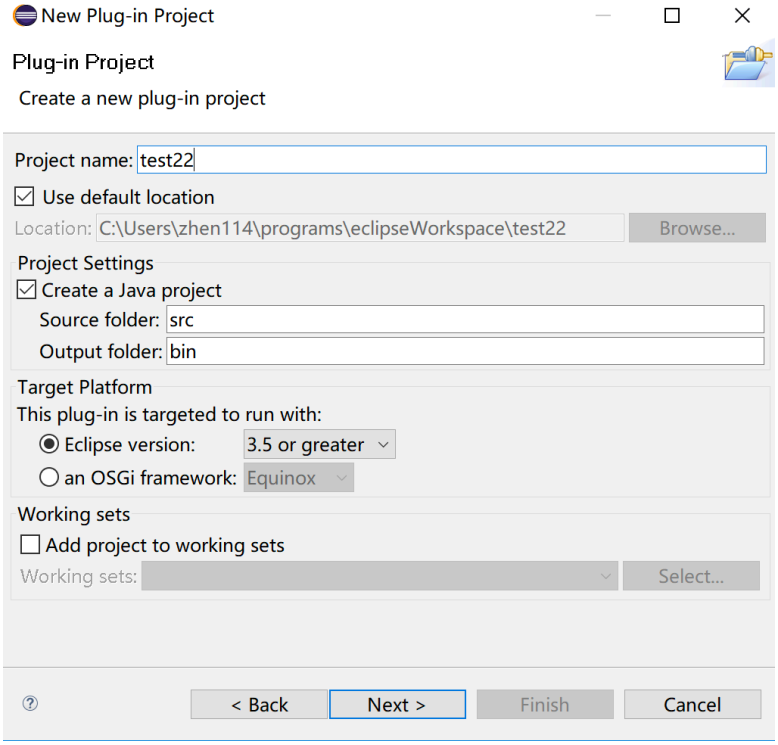
In the menu of Eclipse, File -> New -> Project




Choose Plug-in Development -> Plugin Project

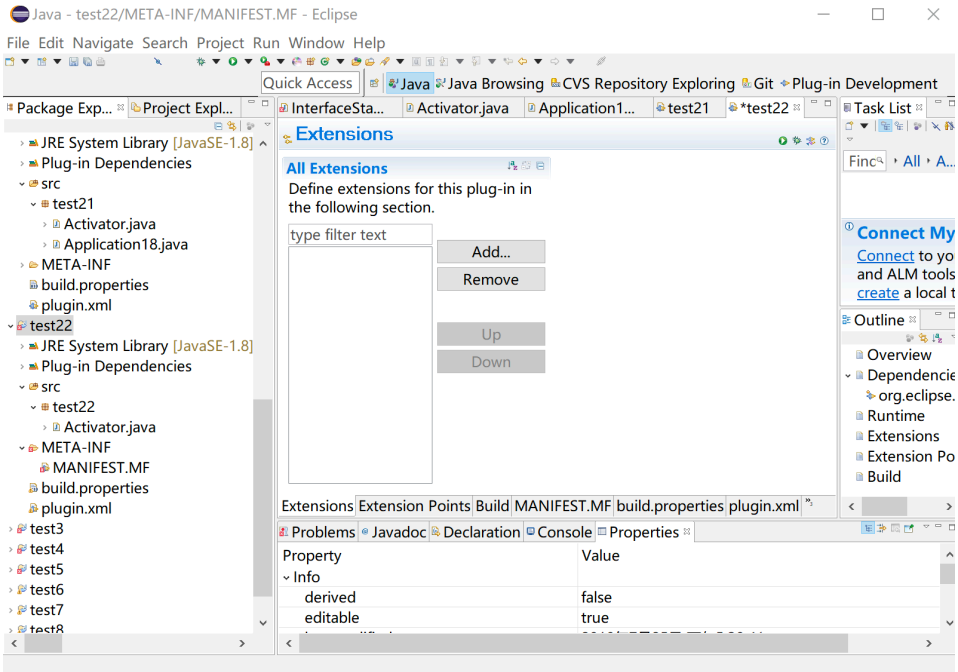


Input the project name and basically just click “next”, and finally we can create a new empty plugin project

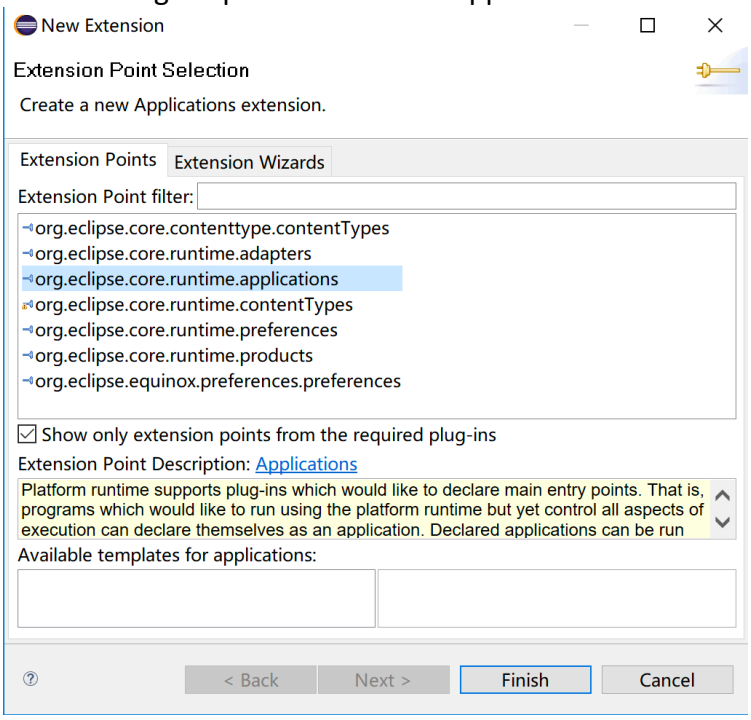



## 2. Make the dependency and write the code

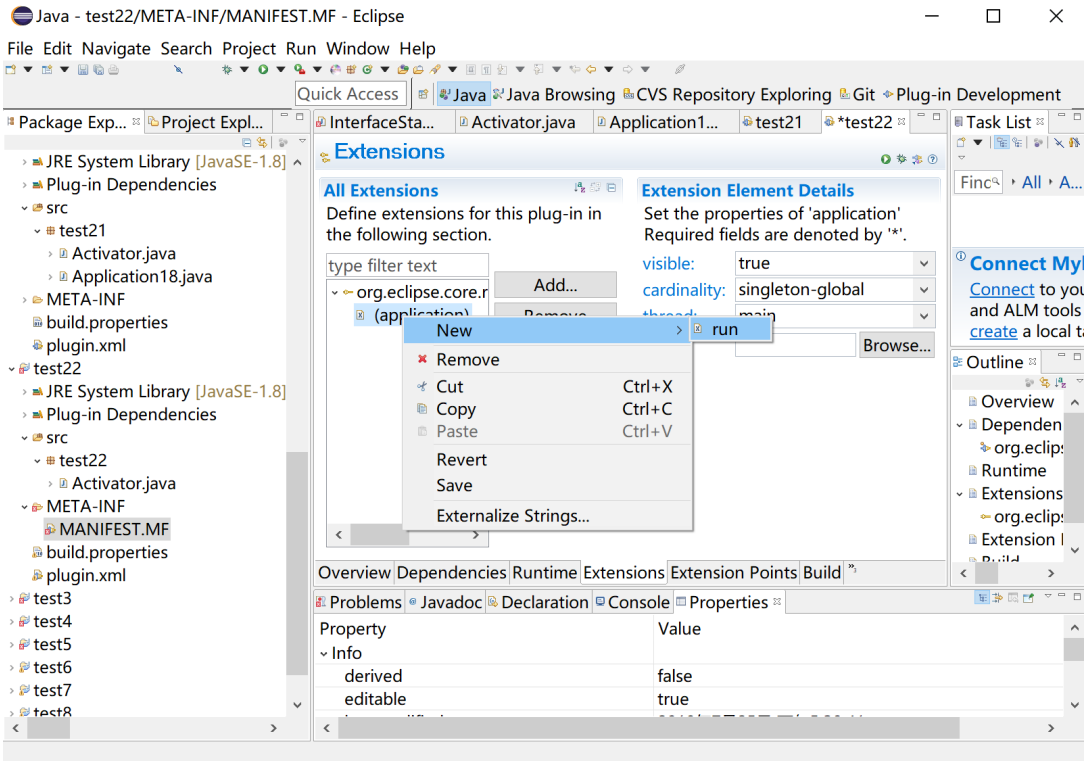
Choose the “Extensions” tab (you can get into this part via multiple options, such as META-INF -> MANIFEST.MF, or build.properties or plugin.xml). Click on the “Add” button.



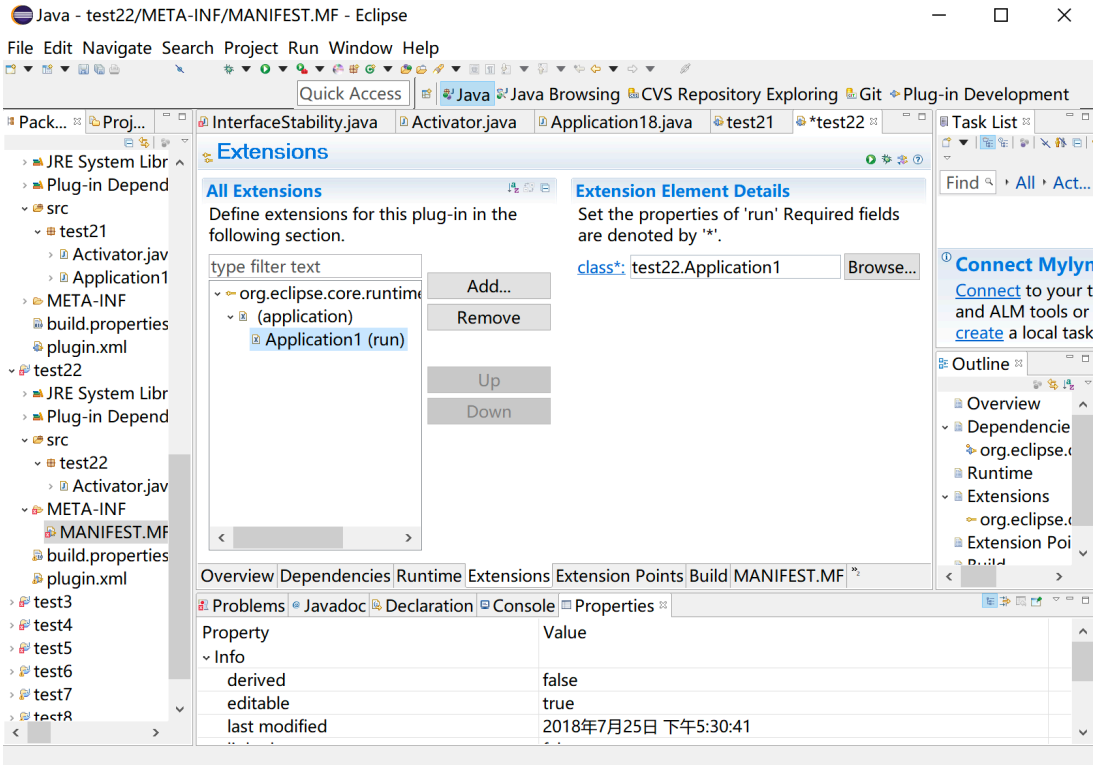
Add the “org.eclipse.core.runtime.applications” and click “Finish”



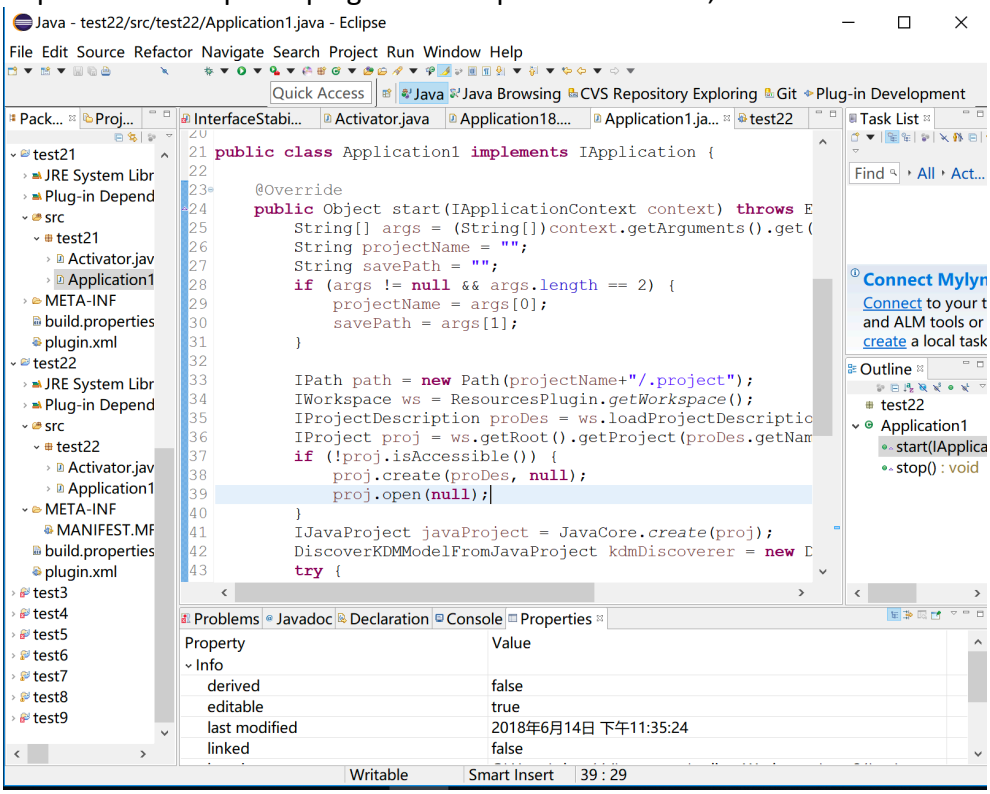

Right click on the new “application” item, New -> run.



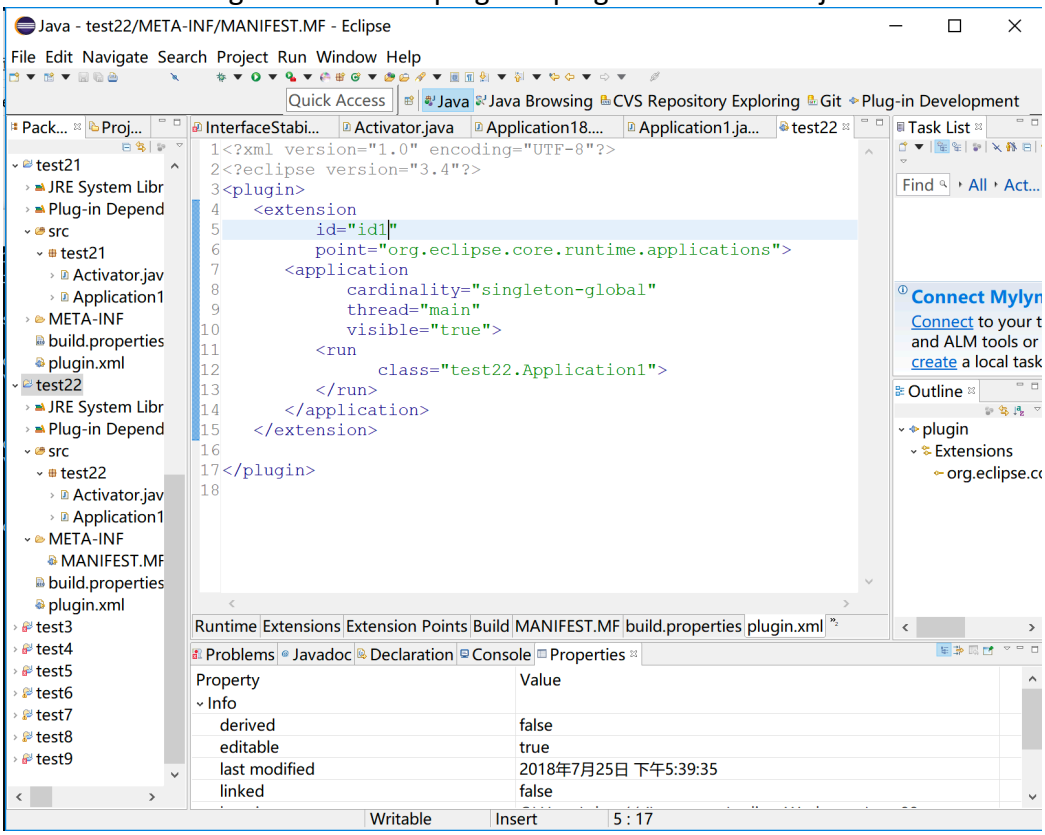
Then we can see the new class option (whose name defaults to be projectName.Application1). Now we click on the “class*” and create the new application class.




Import all the required plug-ins in “Dependencies” tab, and write the code in Application class.

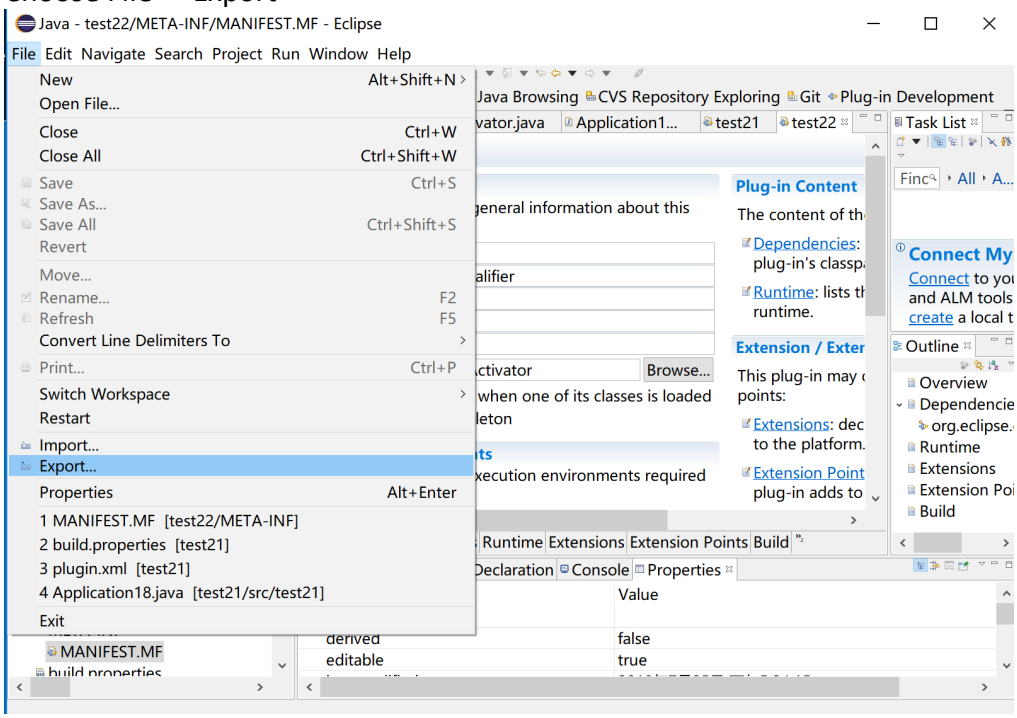


We can also change the id of the plugin in plugin.xml. Here we just use its default value “id1”.

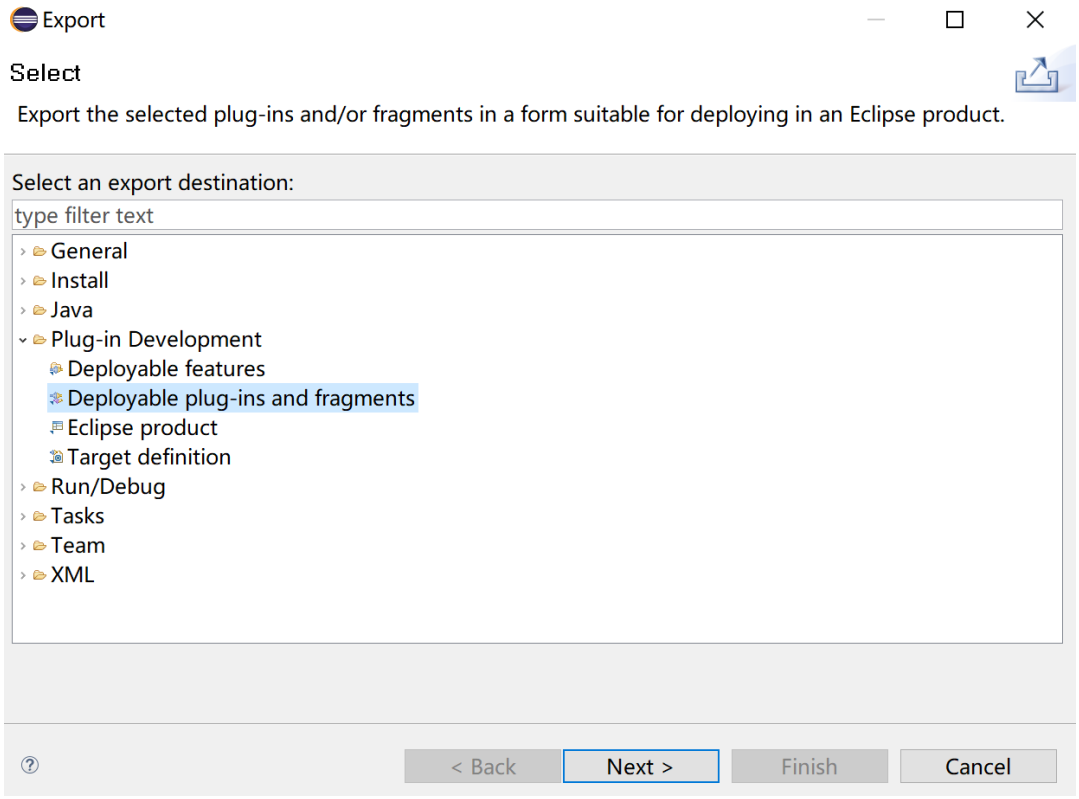



## 3. Export the headless plugin project into a jar file

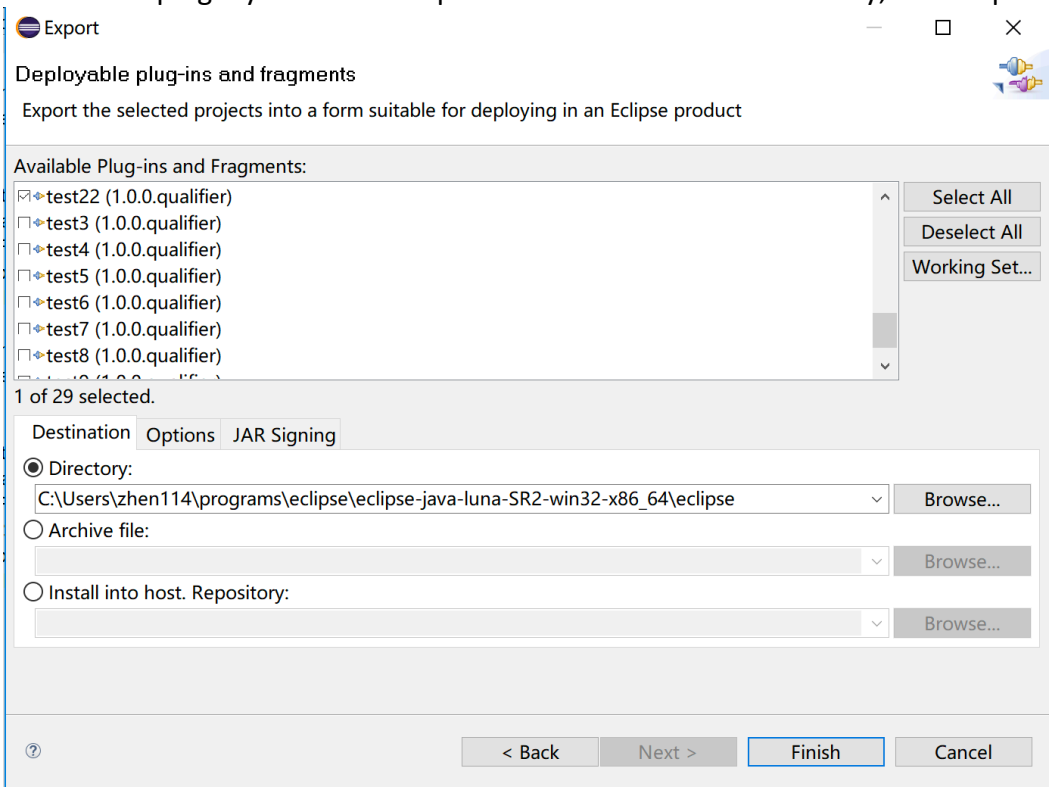
Choose File -> Export



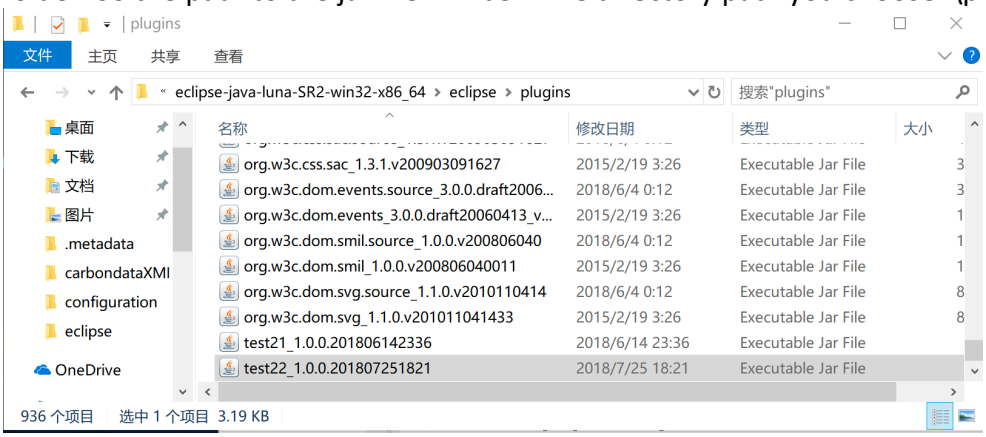
Choose the export destination: Plug-in Development -> Deployable plug-ins and fragments.




Choose the plugin you want to export and the destination directory, then export it.



Then there will be a “plugins” folder in the directory path you choose, and the new jar file will be exported into that folder. If the “plugins” folder already exists, then the jar file will be in that folder. So the path to the jar file will be: “The directory path you choose”\plugins.




If you do not output the jar file in the plugins folder of eclipse, then put it there. Then we can run it from the command line.

### 4. Run the jar file in the command line

To run the jar file, firstly, get into the eclipse root directory in the command line. Then, use the command like this:

eclipse -application projectName.idName <The path to your project> <The path to the output file>

For instance:

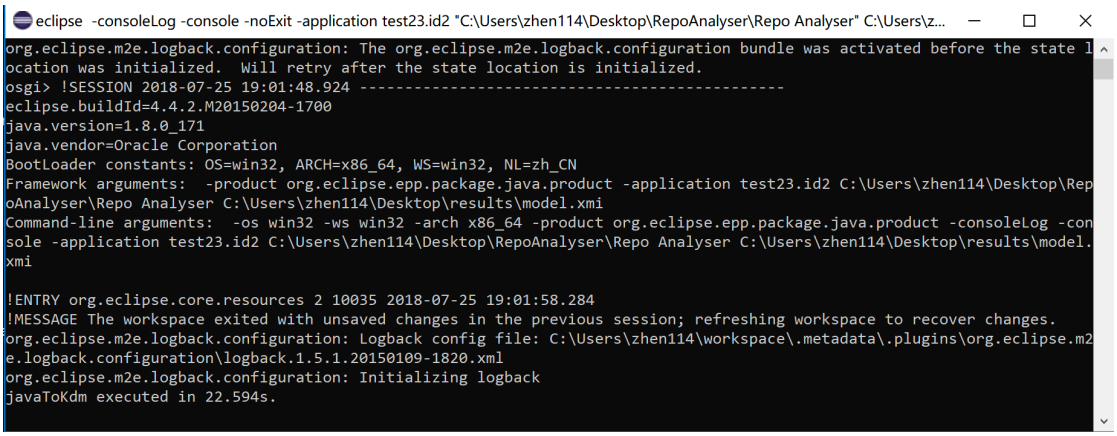
eclipse -application test22.id1 "C:\Users\zhen114\Desktop\RepoAnalyser\Repo Analyser" C:\Users\zhen114\Desktop\results\model.xmi

Note:

1) <The path to your project> actually means the path to the parent folder of the “.project” file in the project you want to convert.
2) If you do not execute your jar correctly for the first time, then after you make some changes in the source code and create a new jar, and replace the old one with it in the plugins folder of eclipse, it may still show the error like “java.lang.RuntimeException: Application "test22.id1" could not be found in the registry”. To solve this problem, you can create a new plugin project, get the jar file, and run it, just don’t make any mistake this time…

You can also use some other option parameters, like -console, which will open the osgi console window, and -noExit, which will keep the console window after the finish of the execution, or -consoleLog, which will show the log in the window.

When the extracting execution is done, you can see “javaToKdm executed in …s”




Then you can find your xmi file.