

## JAVA EE 7 Tutorial – Eclipse

This tutorial is provided for the machines provided by Telecom, if you want to use your own computer, follow this tutorial [here](#) (second answer) but don't ask, we provide you with computers; we're not here to install stuff on yours.

### Safety Checking – Must do

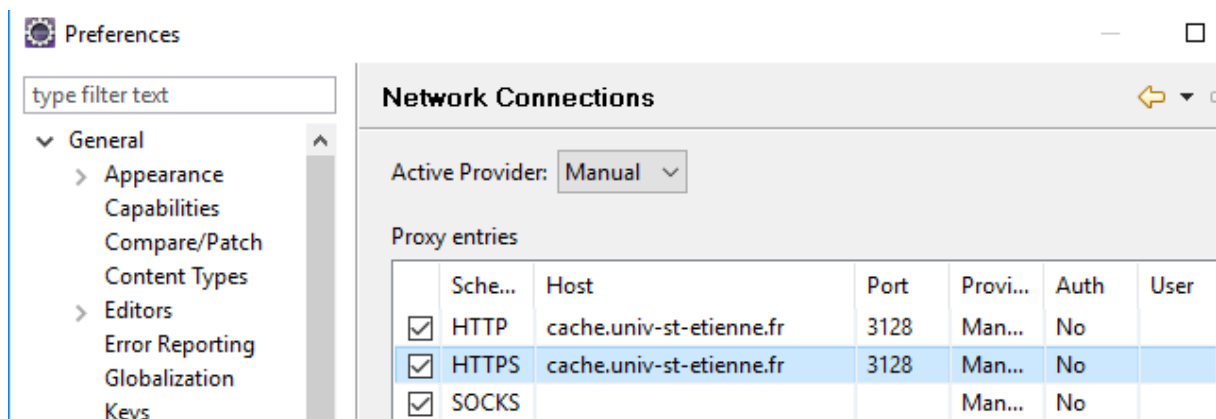
Please be sure all this is correctly done. If you ask a question and I find out you did not follow this part, I'll be very pissed off.

#### Network prerequisites

Our network architecture lives somewhere in the past and we are using a non-transparent proxy that is a huge when our IDE requires an internet access.

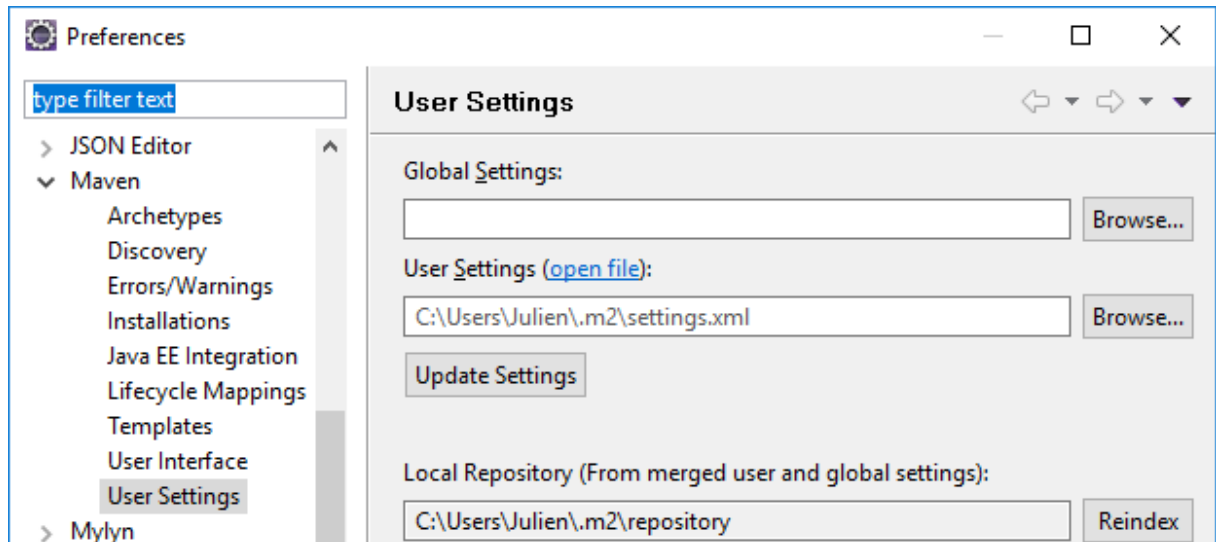
So you must ensure that the following items are well configured:

Proxy connection for eclipse



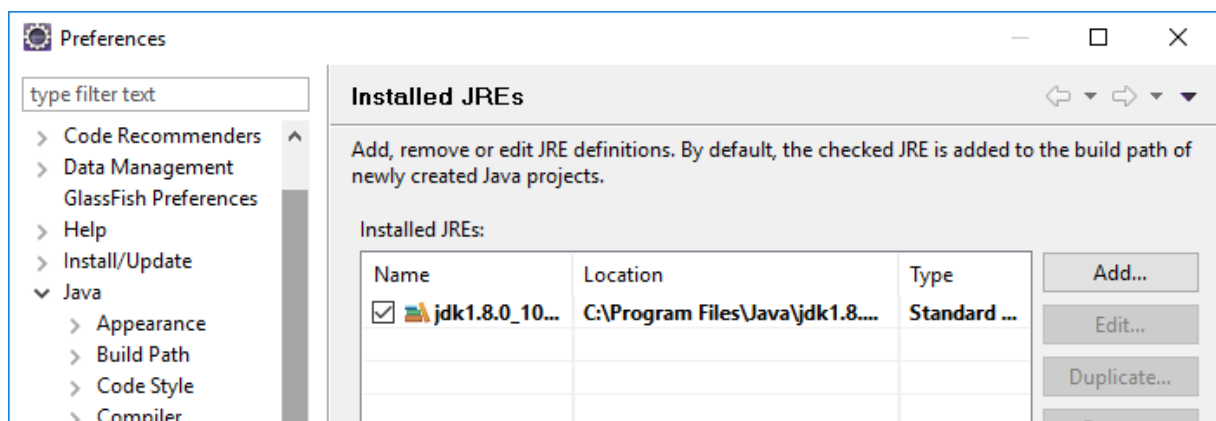
## Maven settings

If no settings.xml is set, get the one from the mootse.



## Java installation

You need a JDK as installed JRE and not a JRE:



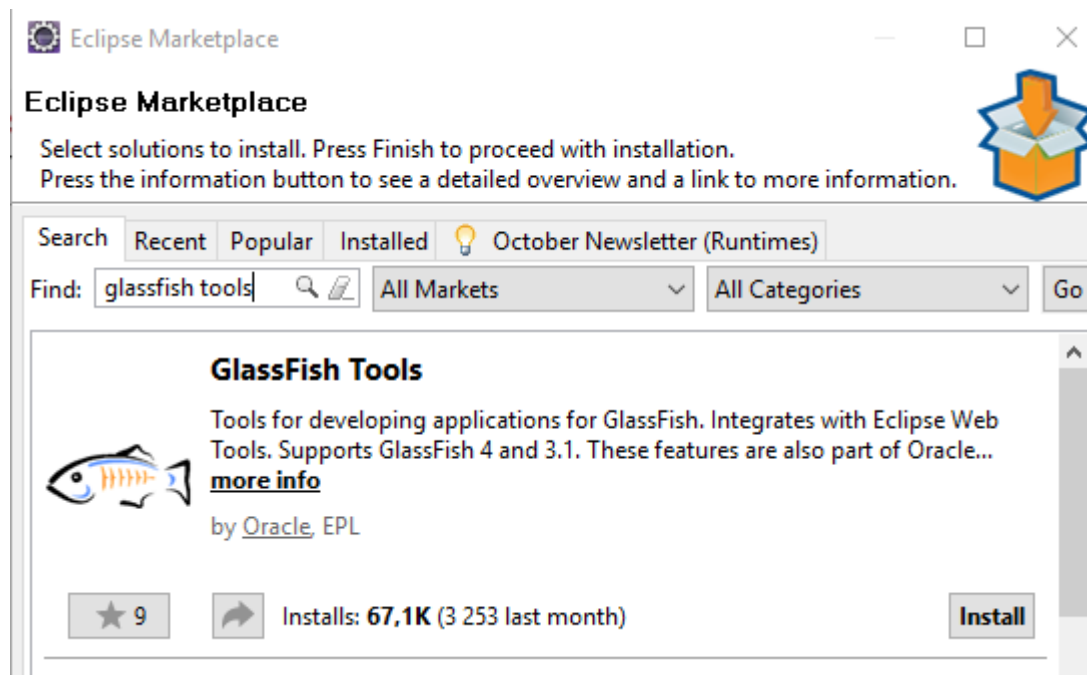
## Download glassfish & samples

Download this [file](#).

Unzip the file on the D drive (you have all the required rights there).

## Install Glassfish tools

Go to Help -> Eclipse Marketplace ... and search for glassfish tools and install it (you can include Java EE 7 documentation).



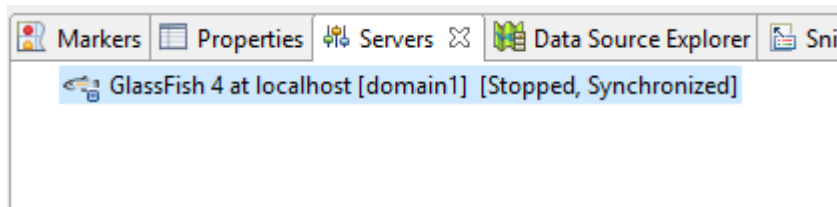
If you cannot connect to the internet, you must setup the proxy (<http://cache.univ-st-etienne.fr:3128>) in the Help -> Preferences -> General -> Network connections, set provider to "active" and fill the required fields. See at the beginning of the document.

## Configure Glassfish from Eclipse

Create a new server (from the server tab, or from new ...), set the directory to D:\glassfish4\glassfish.

If the Java directory is not specified, this means your Eclipse is running a JRE instead of a JDK. Cancel the new server operation and change the installed JRE to a JDK in Help->Preferences-> Java -> Installed JRE and point to the JDK 8. Then recreate a new server.

You should see the following:



## Import the hello-world example

In the project pane, right-click and import existing maven project (type maven to restrict choices), select the one from D:\glassfish4\samples\ejb\hello-stateless-ejb

If you cannot find the sample project, download it from [here](#)

If this is the first import on your machine, this may take a while.

Depending on your configuration, there may be a small bug regarding the Java source level. Open the pom.xml from the *hello-stateless-ejb* project, select pom.xml to see the source and add the following line at the end, just before </project>

```
<properties>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
</properties>
```

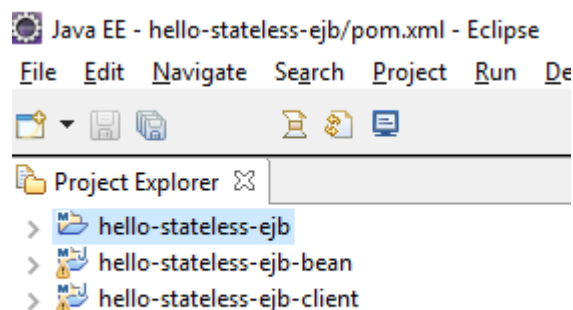
This must look like this:

```
28 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/POM/4.0.0" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/POM/4.0.0" >
29   <modelVersion>4.0.0</modelVersion>
30   <parent>
31     <groupId>org.glassfish-samples</groupId>
32     <artifactId>ejb-samples</artifactId>
33     <version>4.0-SNAPSHOT</version>
34   </parent>
35   <artifactId>hello-stateless-ejb</artifactId>
36   <packaging>pom</packaging>
37   <name>hello-stateless-ejb</name>
38   <modules>
39     <module>hello-stateless-ejb-bean</module>
40     <module>hello-stateless-ejb-client</module>
41   </modules>
42   <properties>
43     <maven.compiler.source>1.8</maven.compiler.source>
44     <maven.compiler.target>1.8</maven.compiler.target>
45   </properties>
46 </project>
47
48
49
```

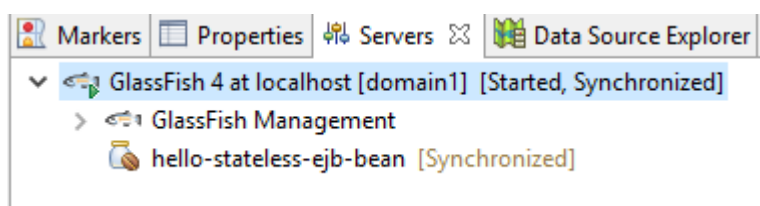
Overview Dependencies Dependency Hierarchy Effective POM pom.xml

Then run maven install on this project (right click, Run as, Maven Install).

Your project explorer must now look like this:



Right click on *hello-stateless-ejb-bean*, Run As, Run on server. Choose your Glassfish4 server (if not existing, then you missed the part Configure Glassfish from Eclipse, go back there, do not pass go, do not collect \$200) and click Finish. Now your application is deployed on the server:



You can now start the main class located in the *hello-stateless-ejb-client* project (StatelessJavaClient.java). This method connects to your application and calls the method hello() on the remote StatelessSession Bean:

```
<terminated> StatelessJavaClient [Java Application] C:\Program Files\J
returnMessage():hello, world!

StatelessSession bean says : hello, world!
```

Inspect the source code of both projects, please note that the StatelessSession Interface is defined in both client and server. On the server side, this interface is annotated @Remote to indicate that it can be called remotely.

## Configure Derby

Glassfish comes with Derby as a database server. While I would not advise to use it in production, this is perfect for this course. Derby is installed in D:\glassfish4\glassfish\javadb

We will first start the database: (replace F: by D:)

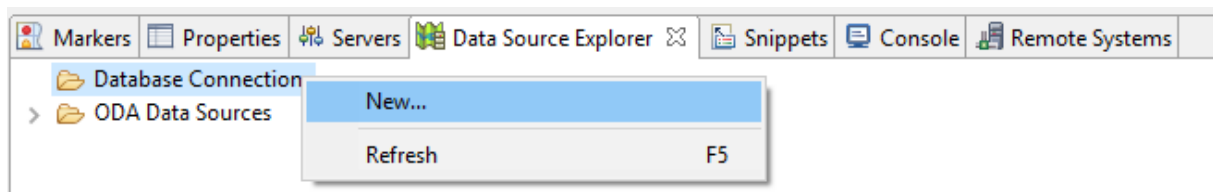
```
C:\Windows\System32\cmd.exe
C:\Windows\System32>f:
F:\>cd glassfish4\glassfish\bin
F:\glassfish4\glassfish\bin>asadmin start-database
```

This will finish like this :

```
Starting database in the background.
Log redirected to F:\glassfish4\glassfish\databases\derby.log.
Command start-database executed successfully.

F:\glassfish4\glassfish\bin>
```

Now we must configure the database within Eclipse:











## Connection Profile

Create a Derby connection profile.

Connection Profile Types:

type filter text

-  DB2 for Linux, UNIX, and Windows
-  DB2 for i5/OS
-  DB2 for z/OS
-  Derby
-  Generic JDBC
-  HSQLDB
-  Informix
-  Ingres

New Derby Connection Profile

**Specify a Driver and Connection Details**

Select a driver from the drop-down and provide login details for the connection.

Drivers: Derby Client JDBC Driver

Properties

General Optional

Database: sample

Host: localhost

Port number: 1527

User name: user

Password: ...

☒ Create database (if required)

☐ Upgrade database to current version

☐ Save password

URL: jdbc:derby://localhost:1527/sample;create=true

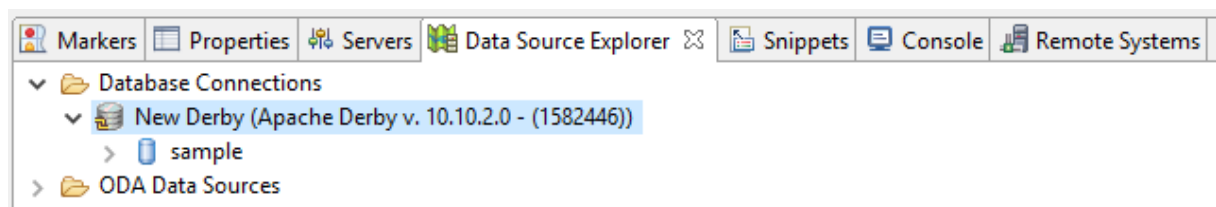
☒ Connect when the wizard completes

☐ Connect every time the workbench is started

Test Connection

< Back Next > Finish Cancel

All the fields are pre-filled, just press finish



## Common errors in importing other samples project



You should try the different samples provided with Glassfish. I would advise to start with importing CDI->transactional.

Anyway you will probably get the same error all the time regarding the JAVA level. Add at the end of the pom.xml before </project>

```
<properties>  
    <maven.compiler.source>1.8</maven.compiler.source>  
    <maven.compiler.target>1.8</maven.compiler.target>  
</properties>
```

Run Maven Install

Click on project then ALT+F5 (Or Right-Click-> Maven -> Update project) this should solve the issue.