Configuring Hibernate with Java Web Project Using JAG(Java Application Generator) :

1. **First run JAG(Java Application Generator).**

```
1. go to : path to JAG/bin/
2. for windows:
    double click gui.bat
   for linux like systems:
    open terminal.
    write command: ~$ cd path to JAG/bin
            /jag-6.1/bin$ chmod +x gui.sh
        /jag-6.1/bin$ ./gui.sh
```

>> your JAG will run now.

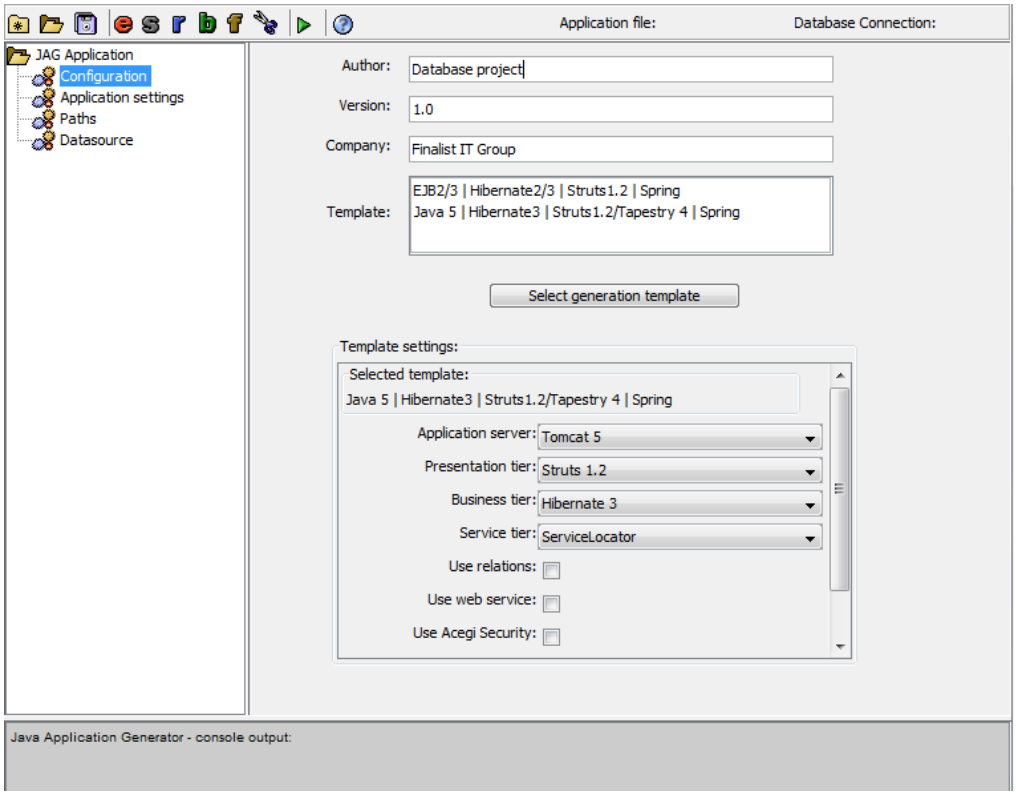
2. Configure your Jag or open saved configuration.

To Configure follow these steps:

1. **Select configuration.**

Do edit >>

Author : Write down a name. --> `RPS team`



`Fig: Configuration`

other fields keep unchanged.

2. Select Application Settings.

Do edit >>

Application name : "give a application name" `rps`

Root-package : edit "com.finalist" to "yourPackageName". `rpsHibernate` (!RPS)

Logging : change "log4j" to "jdklogging"

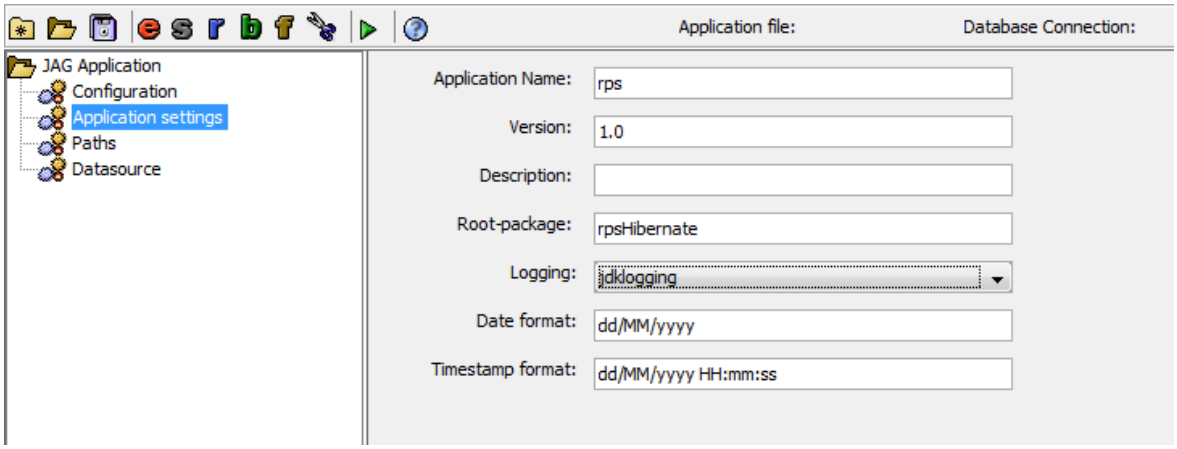


Fig: Application Settings

other fields remain unchanged.

3. Select Paths.

NO NEED TO CHANGE.

4. Select Datasource.

Do edit >>

JNDI name : jdbc/"any name" `jdbc/RPS`

Database type: -->MySQL

JDBC url : jdbc:mysql://<host>/<database>

`jdbc:mysql://localhost:3306/sust_admission`

here....

<host> is "localhost:3306" for localhost database.

<database> is "Schema name of Database"

example: --> jdbc:mysql://localhost:3306/sust_admission

user name : "database user name" (ex. "root") `root`

password : "password for the database" `sust`

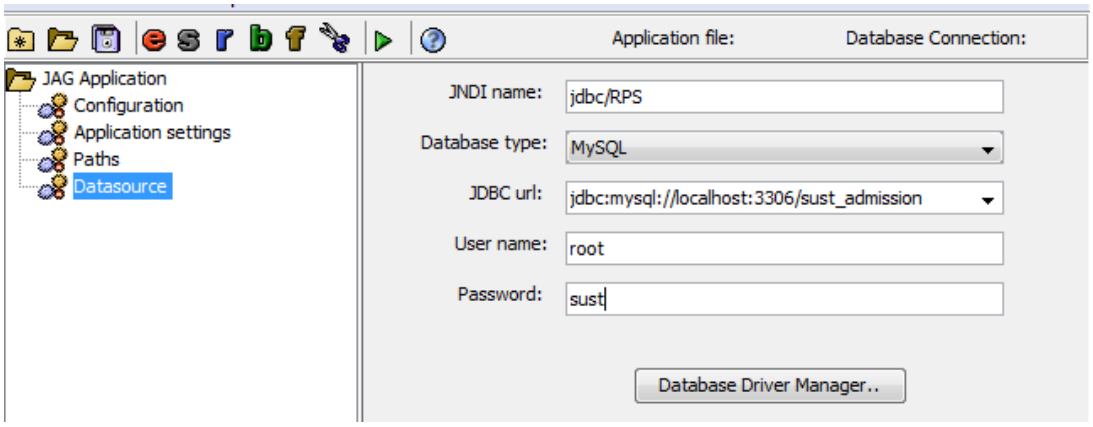


Fig: Datasource

now you can save your configuration by clicking save.

6. Select "**new entity bean**" from the tool bar (sign "e").

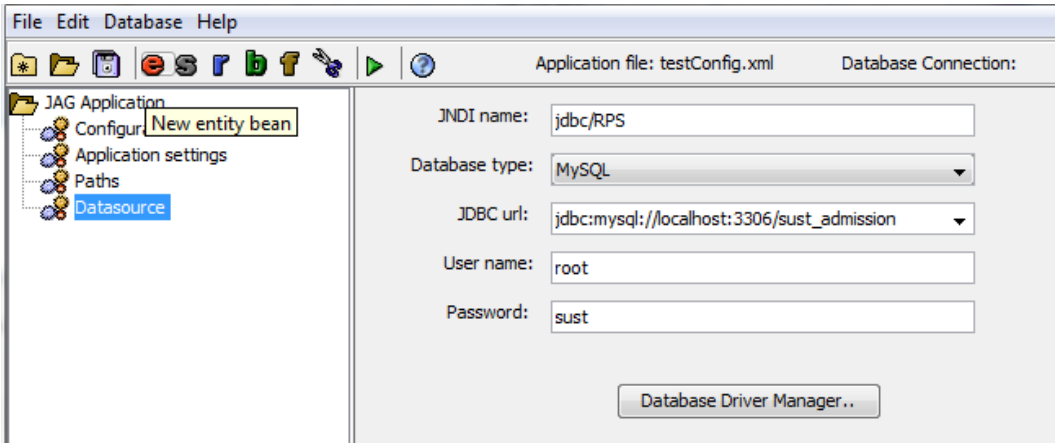


Fig: Select New entity bean from toolbar

After clicking following window will open:

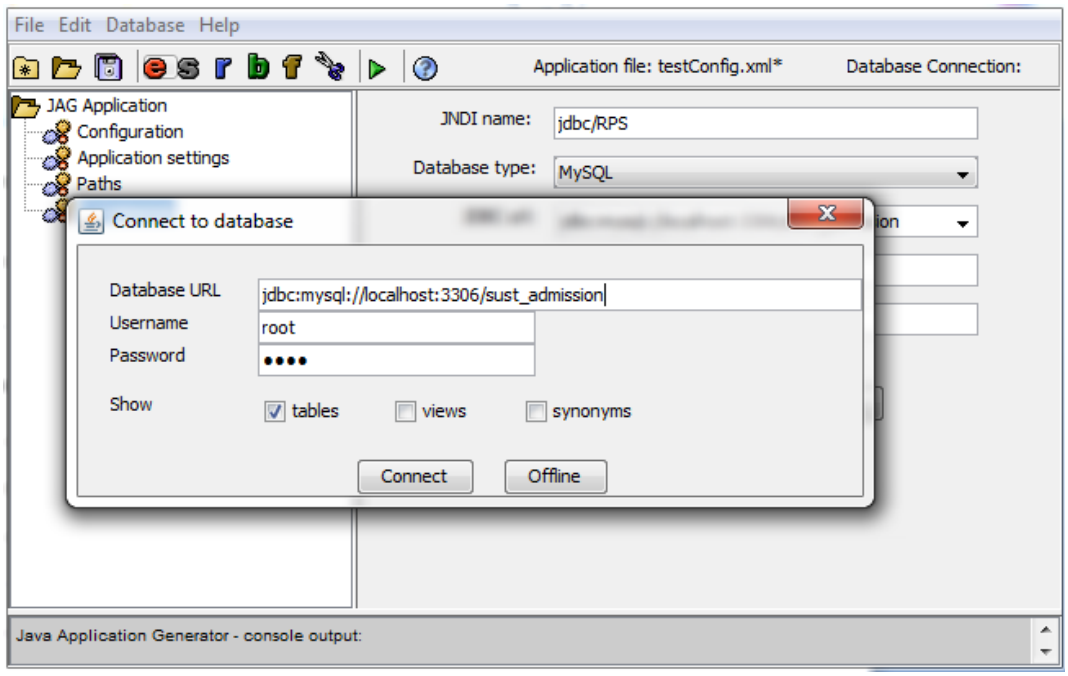


Fig: Generating entity beans

click "connect"

Select table names from the list. -->select all the table (Ctrl+'A') or selective tables and click select.

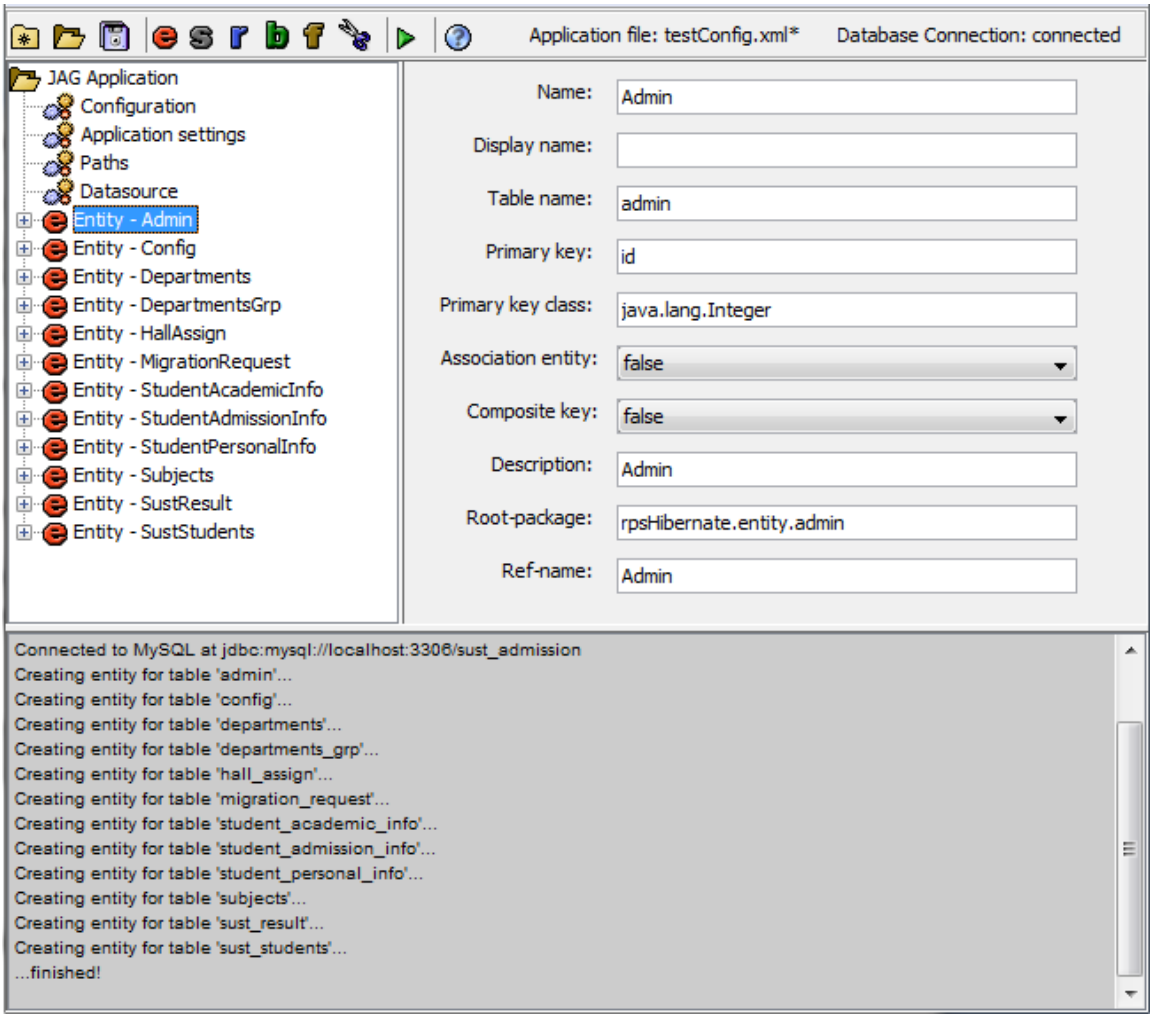


Fig: After generating entity classes

Make following change:

Do edit >>

Display name: "any custom name" --> Copy and paste Name: (the above field of display name)

Root-package: change "rpsHibernate.entity .name" to "yourPackageName.entity"

1. `rpsHibernate.entity` [this will be replacing "rpsHibernate.entity.studentbatch" with "rpsHibernate.entity"]

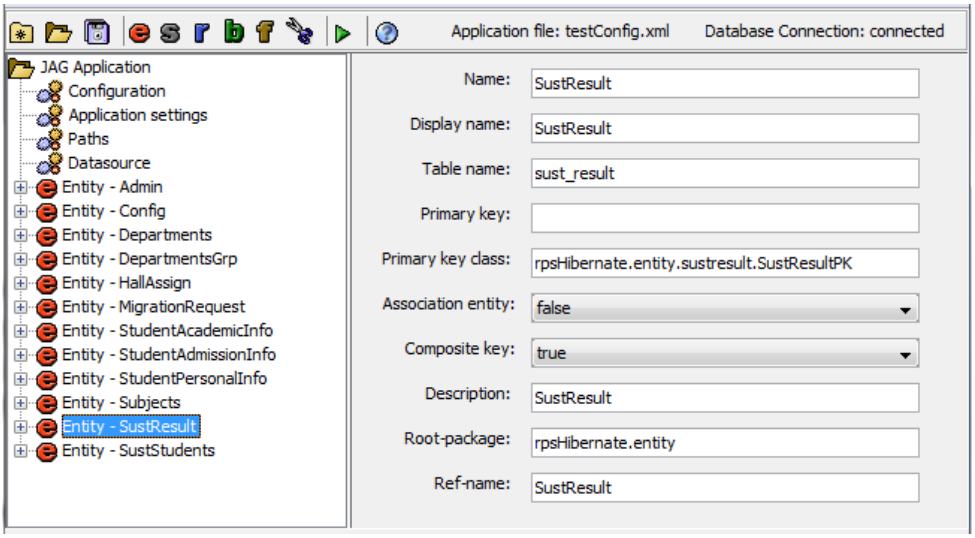


Fig: Edit entity classes

Other field remains unchanged.

***__Complete this process for all entity class, then start session class__

7. click "**new service bean**(sign "s")" from toolbar.

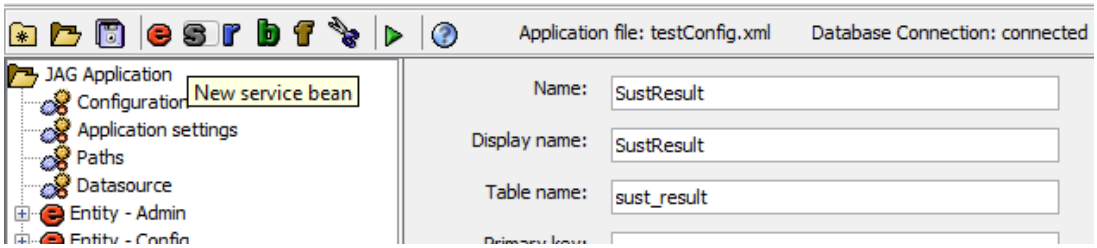


Fig: New Service bean from toolbar

Name : Give a service name.(ex. AdminService) --> here table or entity name (Copy it from entity) appended with String "Service"

Root-package: "yourPackageName.session" --> `rpsHibernate.session`

Refs. : click "add" and add your necessary tables. --> select the table name

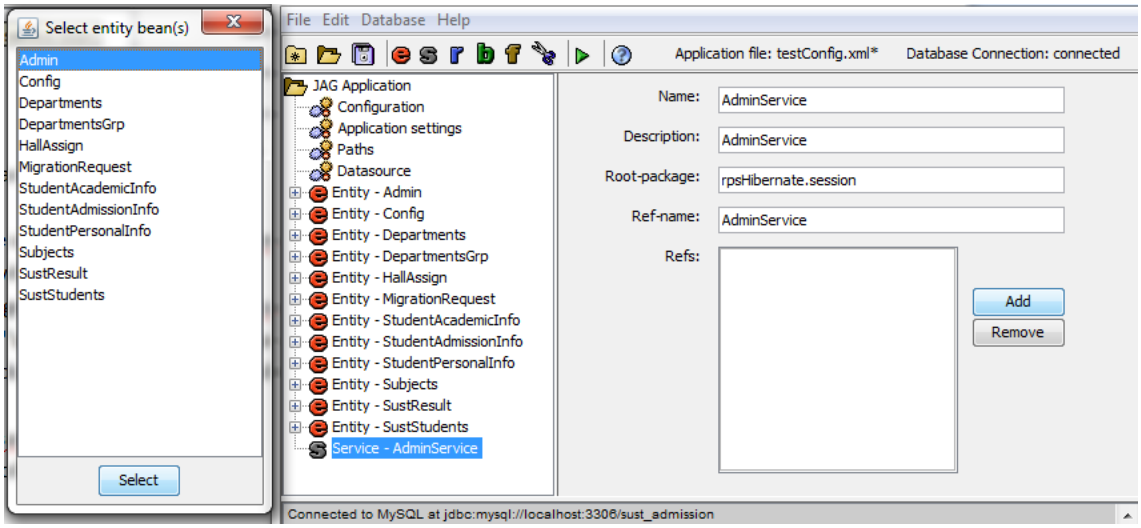


Fig: Configure service class

--> Do this total process for all the tables. Create one service class for each table.

8. Click Save. (sign 'saveicon')
9. Click Run from toolbar. (sign ">")

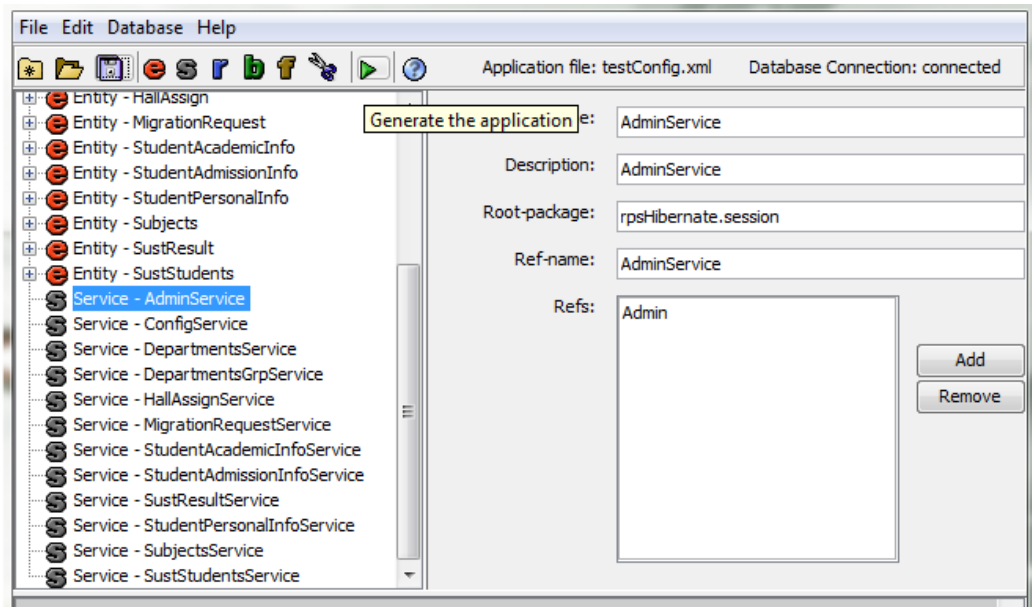


Fig: Run the application

An invalid configuration warning will appear.

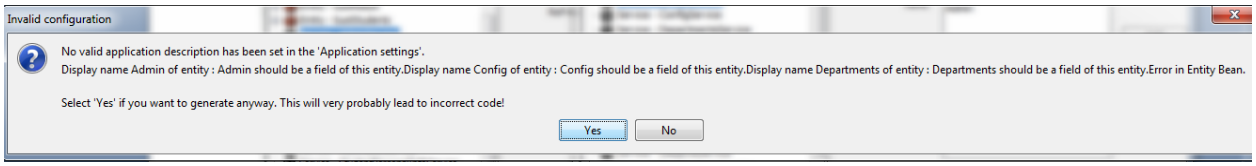


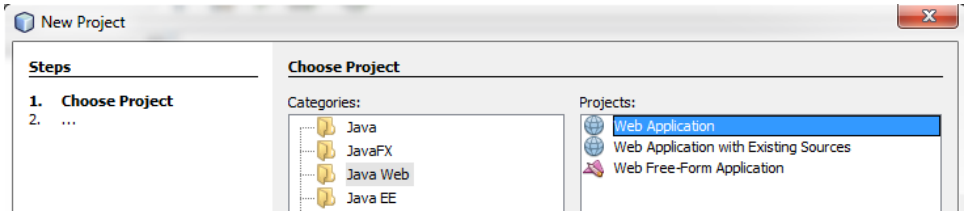
Fig: Invalid warning

Click yes.

*** A folder named ApplicationName will create with all necessery files.

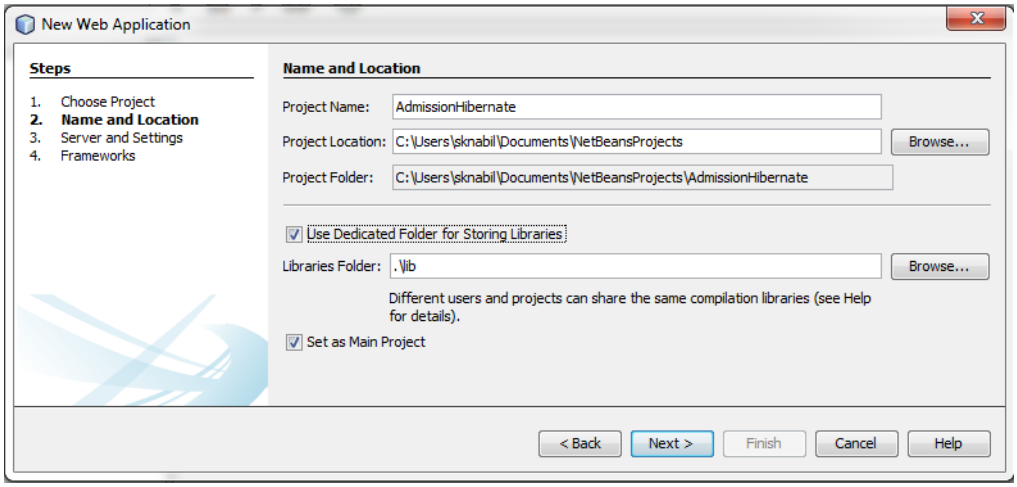
Netbeans process :

1. Choose "new project > java web > web application"



click "next".

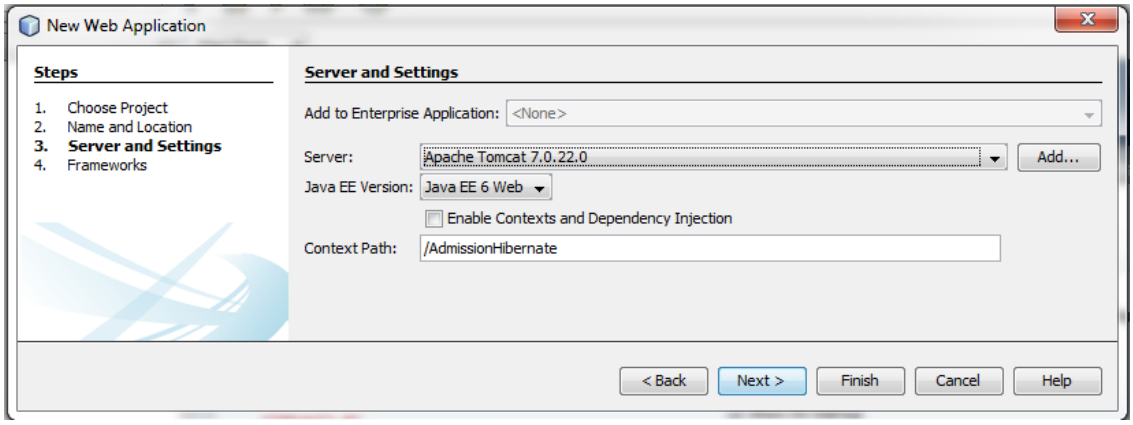
2. Project name : "YourProjectName" , check "use dedicaded folder for storing libraries".



Check "Use Dedicaded Folder for Storing Libraries".

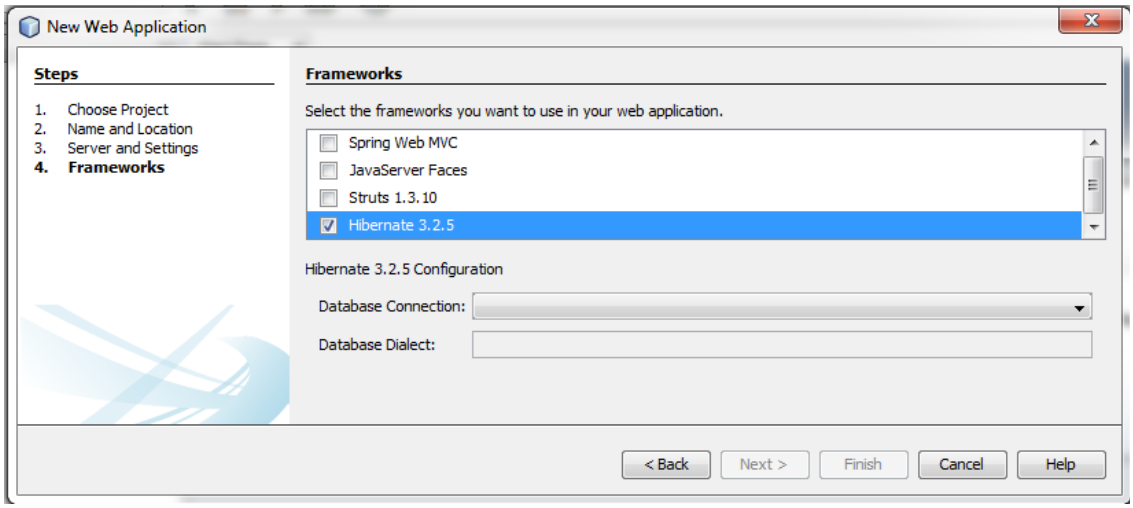
Click "next".

3. Choose your server.

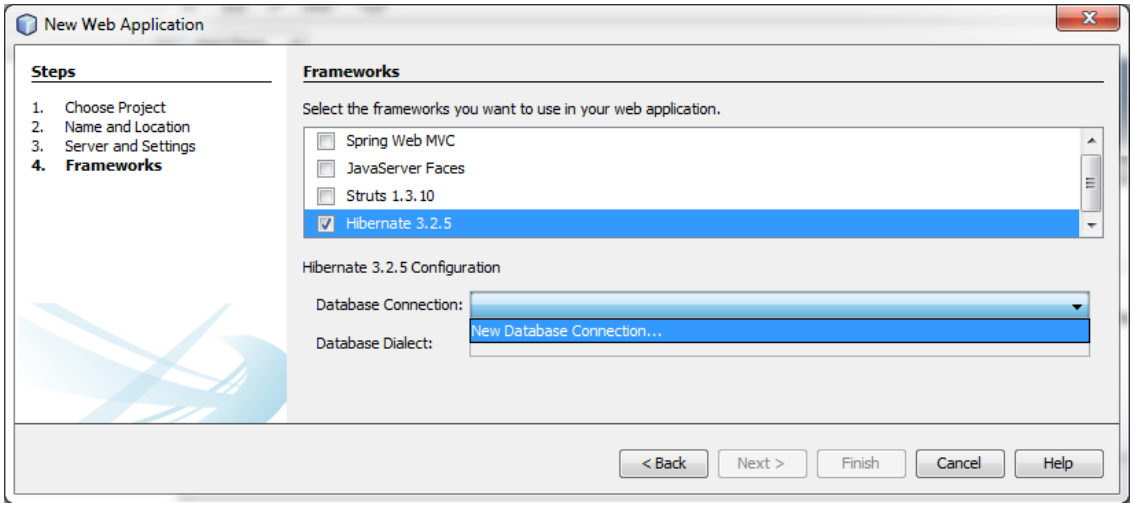


Click "next".

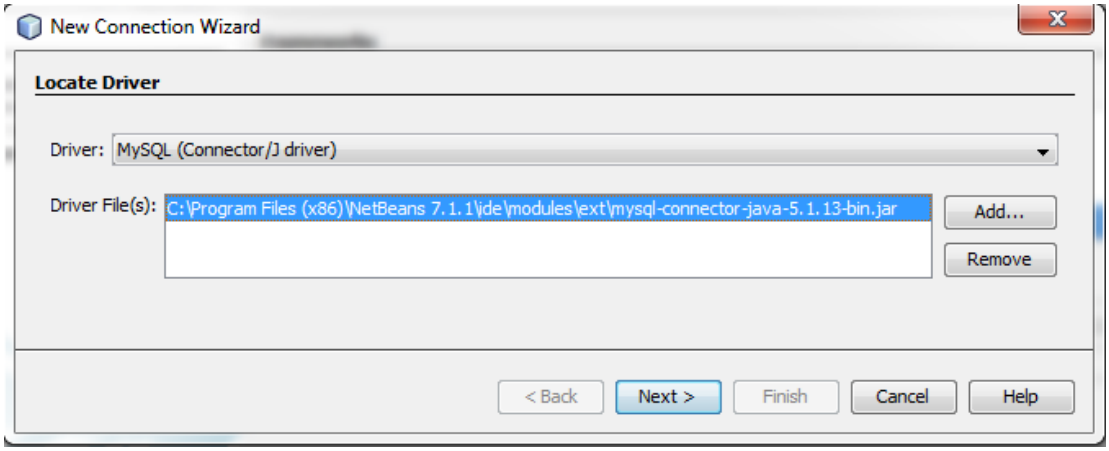
4. Choose hibernate as your framework.



5. Choose "Database Connection". if new connection required choose "new database connection".



(A new window will appear)



Click Next.

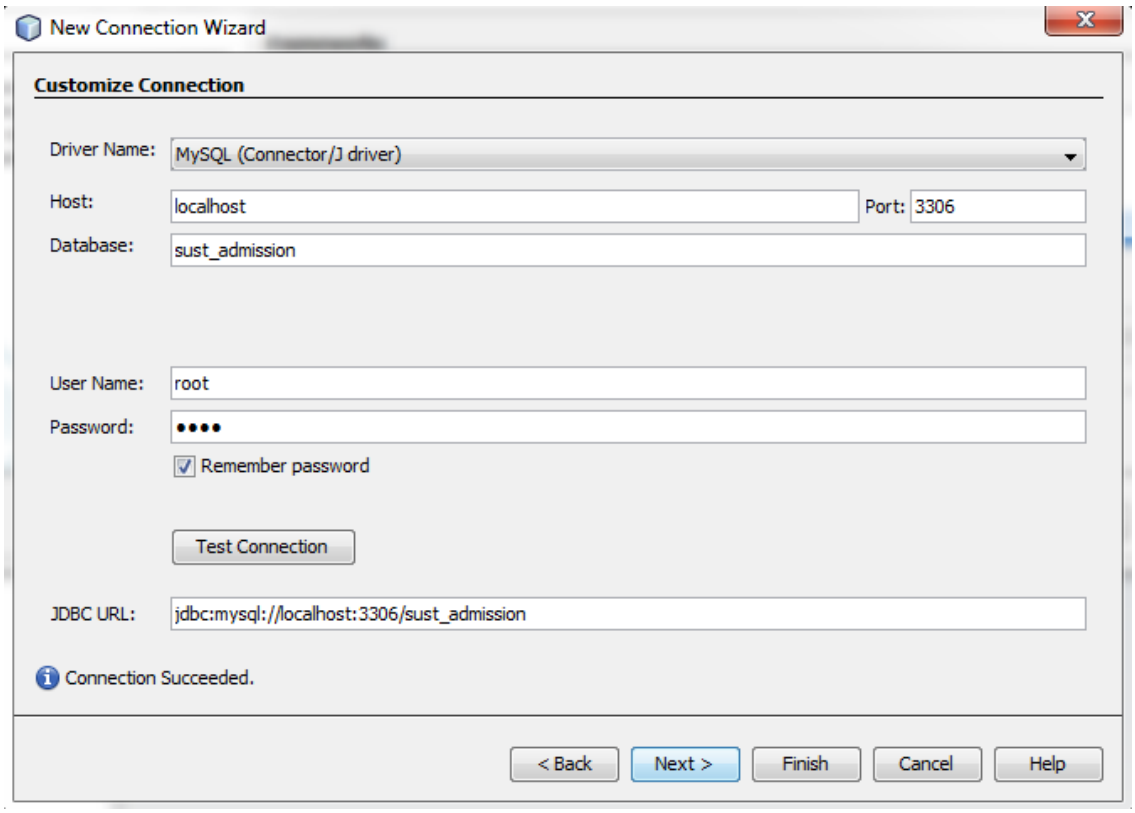
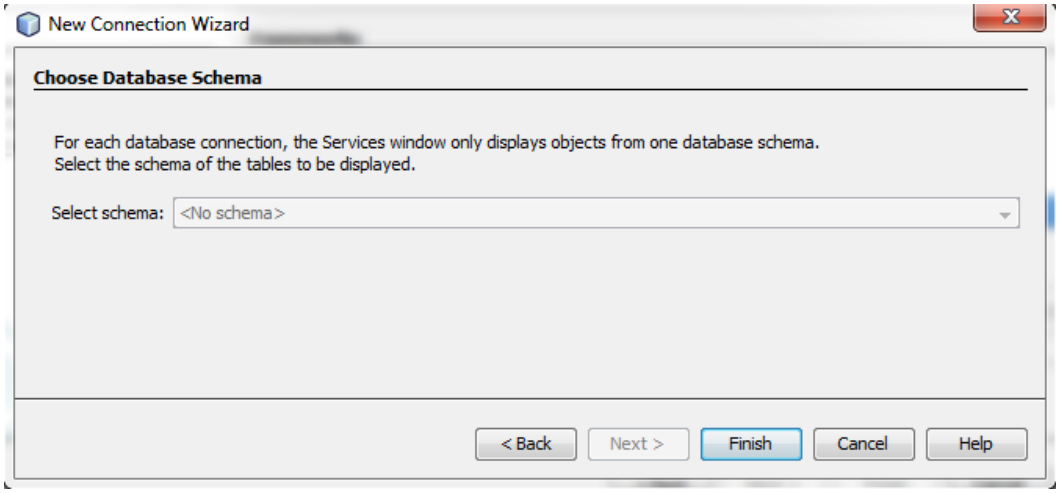


Fig:

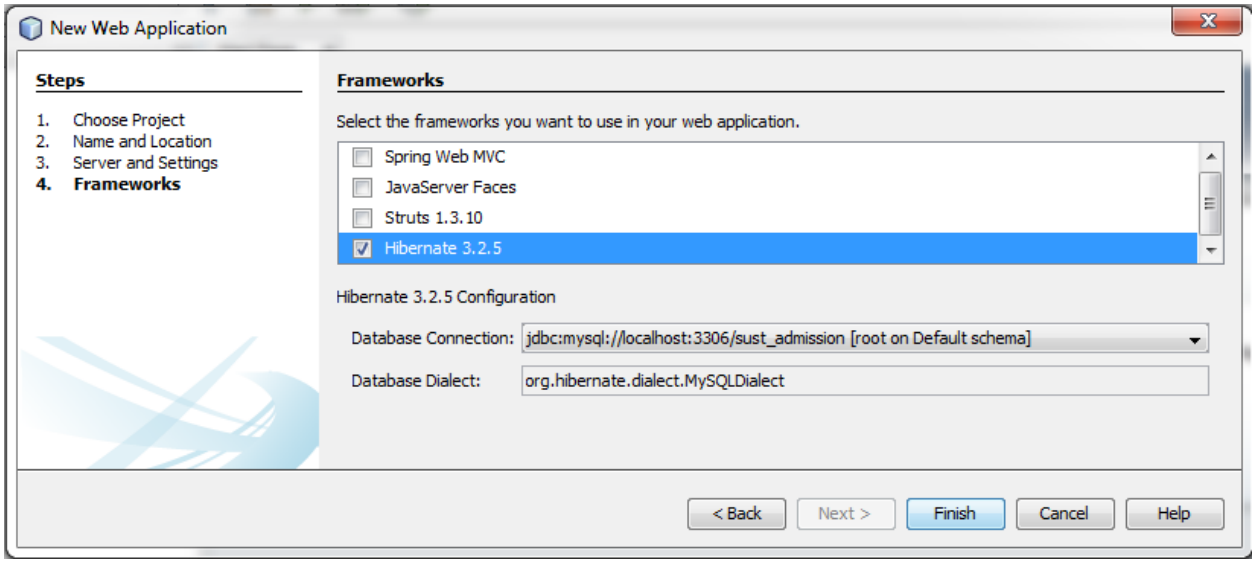
Driver name : "mySql" --> `MySQL (Connector/J driver)`
host : localhost (for localhost database)
port : 3306 (for localhost database)
Database : Database Schema name.
User name : Database user name.
Password : Database password. Check "remember password".

// check "show JDBC url".

Click "Ok".



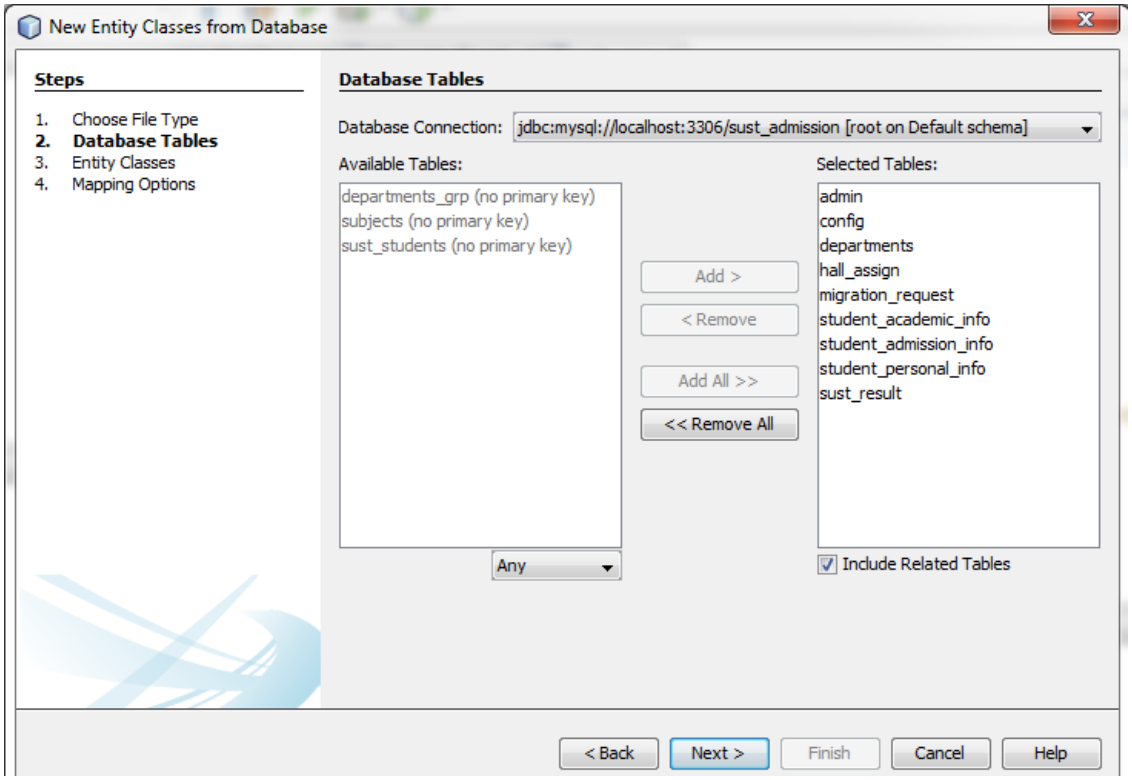
Click "Finish".



Now, again click finish.

After creating web project, generate entity classes using Netbeans.

7. Right click on projectName > new > entity classes from database.



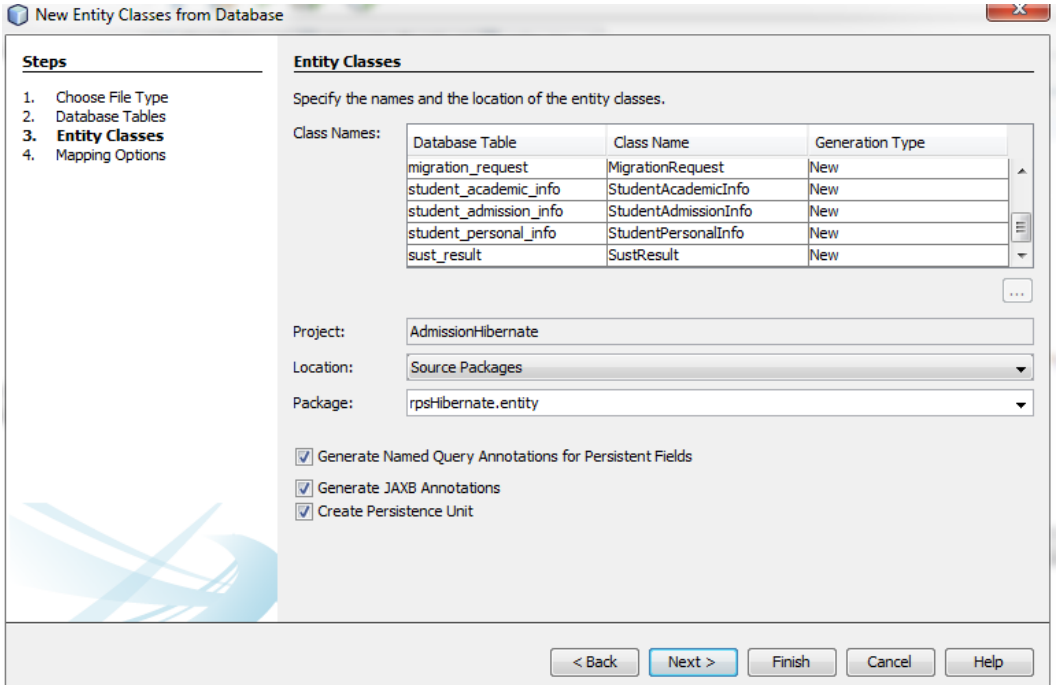
Database connection : select database connection (which you created earliar).

Then, select necessary tables from available table list and click add. --> add all or selective table

Click "next".

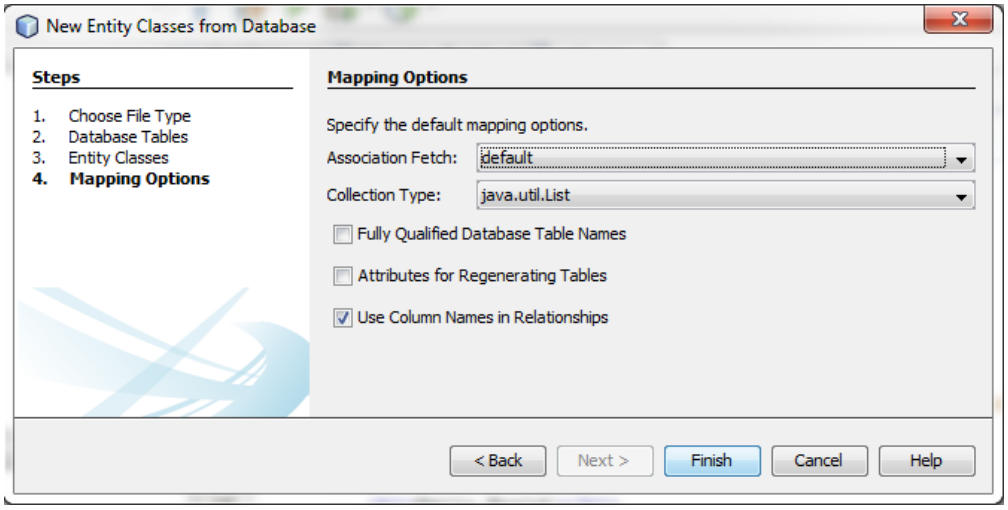
8. Package : "yourPackageName.entity" --> write "rpsHibernate.entity"

check "Generate named query Annotations for persistent fields".



Click Next

9. Change collection type to “java.util.List”



[--> Step 13-16 not Found,,, (not necessary)

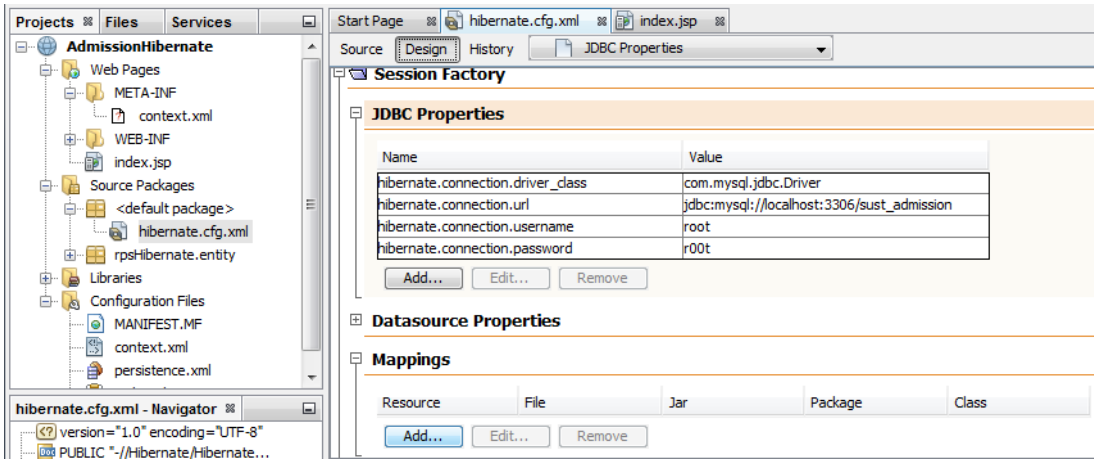
13. Click "Create persistent unit".
14. Change "Persistence library" to "hibernate".
15. Select table generation strategy : "none".
16. click "Create".

]

10. Click "Finish".

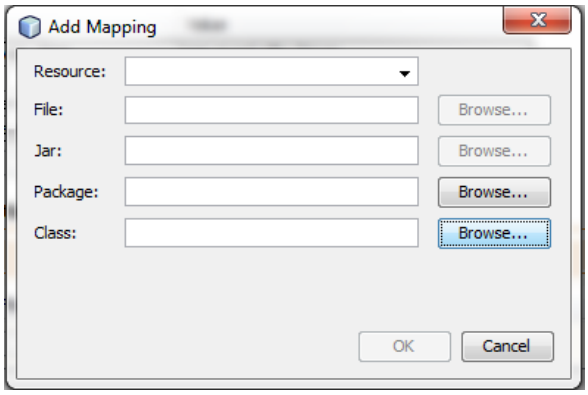
This will automatically generate the entity class from database.

19. Go to hibernate.cfg.xml

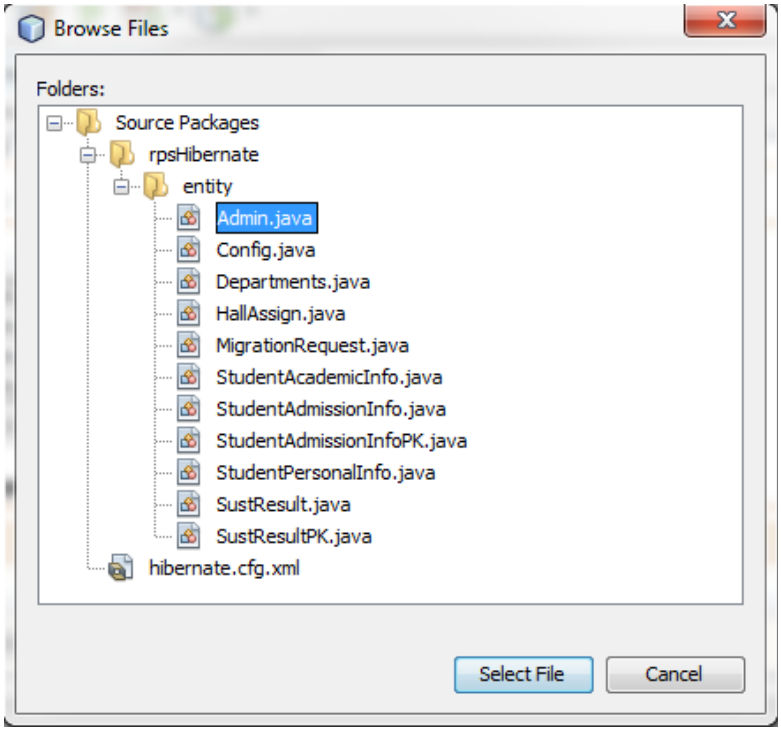


open "Mappings". click "add".

Go to "Class" field. click "Browse".



go to "Source package > Package > entity > entity classes" for all necessary table.



click 'select File'

-->Do this process for all tables.

add these three lines in xml before <mapping class="yourPackageName.entity.class"/>

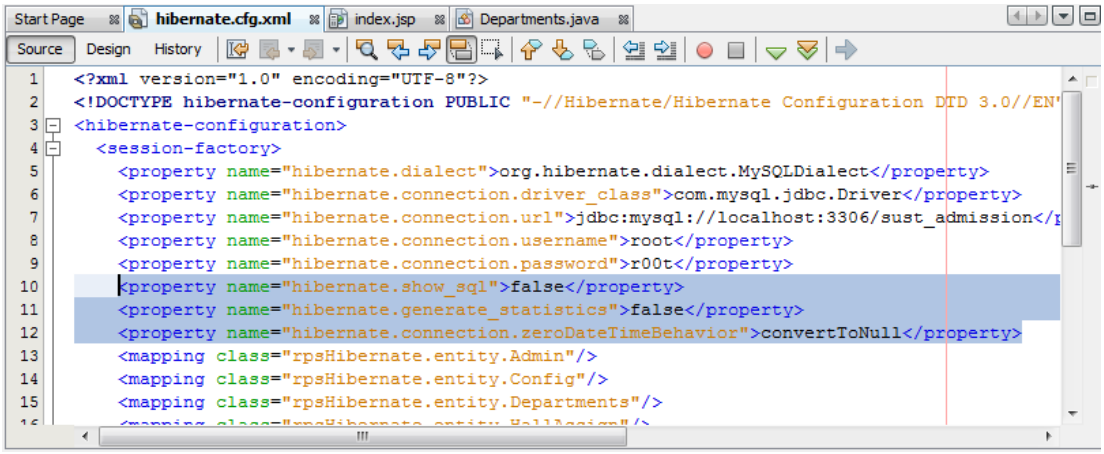
```
<property name="hibernate.show_sql">false</property>
 <property name="hibernate.generate_statistics">false</property>
 <property name="hibernate.connection.zeroDateTimeBehavior">convertToNull</property>
```

--> click Source, paste the above three line in between

<property name="hibernate.connection.password">sust</property>

<<paste the codes here>>

<mapping class="RPS.entity.Assistant"/>



20. Go to project folder generated by JAG. go to source > java-web. Copy "com" folder to /netbeans-project-folder/src/java/ and paste.

Now, delete all files except

"CustomLevel.java",
"JDKLogger.java",
"LogService.java",
"logger.java"

in "NetBeansProjects/YourProjectName/src/java/com/finalist/util/log"

--> Delete all other files in com folder [Just only 4 above class in com folder]

Now (here) the "JDKLogger.java" should be replaced by given "JDKLogger.java" (Find this in your resource).

21. go to project folder of JAG and go to /src/java-web/yourPackageName/ and copy "exception", "servicelocator", "session" folder to /netbeans-project-folder/src/java/yourPackageName/ (where entity is already exist).

24. Create folder "hibernatehelper" in \netbeans-project-folder\src\java\yourPackageName.

Copy "hibernateQueryHelper.java" from /src/java-web/yourPackageName/ of JAG generated folder.

Also copy here "HibernateSession.java" which will be given (Find this in your resource).

//26. Delete any files from "\Hibernate\HiberTest\src\java\com\finalist" [Constants.java]

//27. delete everything except log folder from "\Hibernate\HiberTest\src\java\com\finalist\util"

rename the Forlder --> \NetBeansProjecs\RPS\src\java\RPS to java\rpsHibernate

28. replace “session = rpsHibernate.HibernateSessionHelper.getHibernateSession();” line in 38 no line of hibernatequeryhelper.java with the following given line:
session = HibernateSession.getSession();

also comment 6no line import.

**--> In every entity class, add This method, for example, Assitant.java

```
public java.lang.Integer getPrimaryKey(){
       return getAssistantPk();
}
```

import or paste this three lines to all service class: //This is not necessary

"import rpsHibernate.exception.GenericBusinessException;
import rpsHibernate.servicelocator.LocatableService;
import rpsHibernate.entity.*;"

Sol (If you face any problem):

1. Add this jar http://jaxen.codehaus.org/releases.html
2. Edit the package name in HibernateSession.java class

package rpsHibernate.HibernateHelper;

--- End of configuration ---

//

when editing database:

1. for deleting a table, first delete all foreing keys for this table from other table, then delete it.
2. For creating foreing key of an existing attribute, delete all the values from the table.. else shows mysql exception.
3. To change one attribute in a table which has no relation above or below, simply recreate it’s entity & service class with jag. Delete the entity class form project. Recreate it using
   rpsHibernate new entity classes form database --> :: follow steps above for this.

Then replace jag created service class with projects service class. If modifies where applied before then merge it with the old one.

3. If new table generated, or a table is deleted, then change configuration
   go to hibernate.cfg.xml open "Mappings" :: follow steps above for this.. add or delete
4. If a table has to change or delete an attribute which has came from another table as foreing key, then only create service class for this table and delete this and other tables entity class & regenarate them.

5. If a table has to change or delete an attribute which has sent to another table as foreing key, then create service class for this table and the tables which has used this attribute as foreign key. Then delete this and other tables entity class & regenarate them.

Comments by me:

** To edit or delete or making any change which may cause change in internal mapping, then all affected tables entity class should be deleted & regenarated by netbeans.

** Service classes generated by jag is only for using the methods. It doen't have any relation with internal mapping.

** If other tables sql format,, (attributes, foreing key etc) is unchanged, then for fixing any internal mapping error, just delete and regenerate entity class by netbeans,, no service class needed.

** For making new methods (if any attributes, foreing key etc changes) jag is used to make service class

Prepared by –

Sheikh Nabil Mohammad
Lecturer, Dept of CSE, SUST