Introduction to Hibernate & Configuration with Eclipse(Helios)

HIBERNATE - Introduction to Hibernate

Hibernate is an open source object/relational mapping tool for Java. Hibernate lets you develop persistent classes following common Java idiom - including association, inheritance, polymorphism, composition and the Java collections framework.

Hibernate not only takes care of the mapping from Java classes to database tables (and from Java data types to SQL data types), but also provides data query and retrieval facilities and can significantly reduce development time otherwise spent with manual data handling in SQL and JDBC.

Hibernates goal is to relieve the developer from 95 percent of common data persistence related programming tasks.

Hibernate is Free Software.

Before installing Hibernate, install Jdk1.5 or higher version, Eclipse (This tutorial uses Helios), and install database MySql

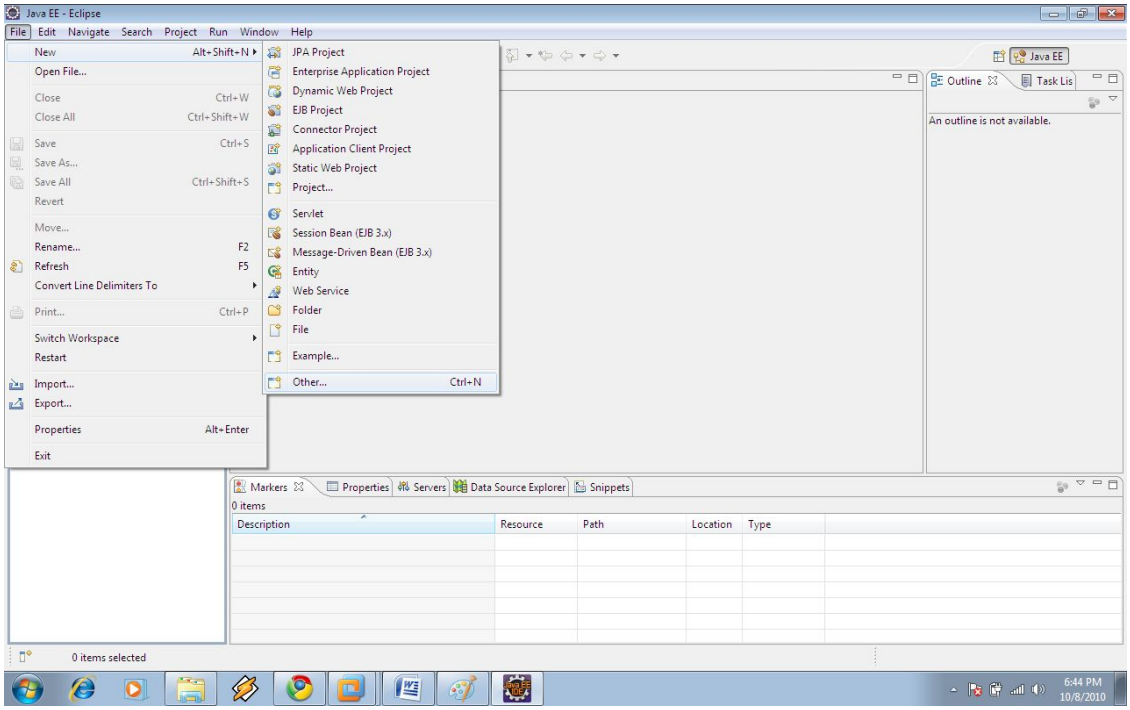
To get Hibernate

Download from **http://sourceforge.net/projects/hibernate/files/**

hibernate-distribution-3.3.2.GA-dist.zip
hibernate-annotations-3.4.0.GA.zip

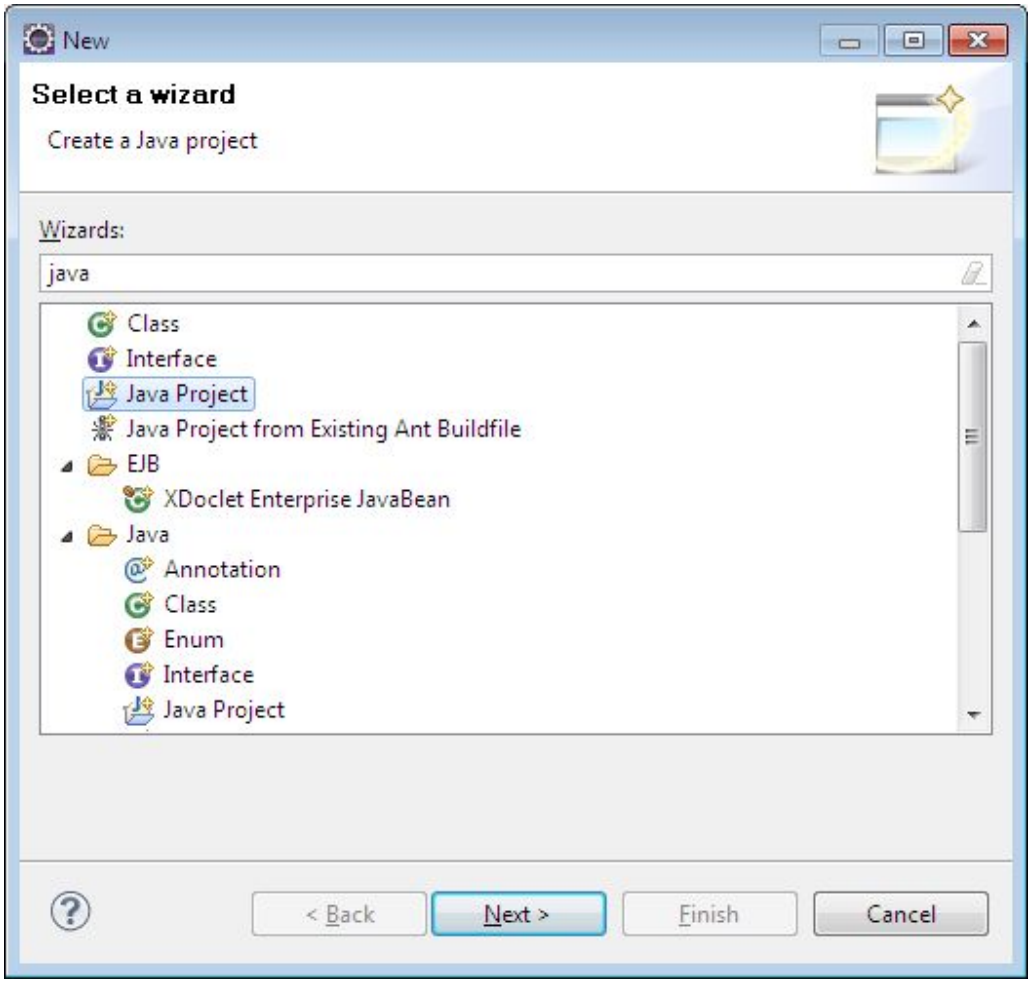
Create a java project
Step 1:

adith.naveen@gmail.com



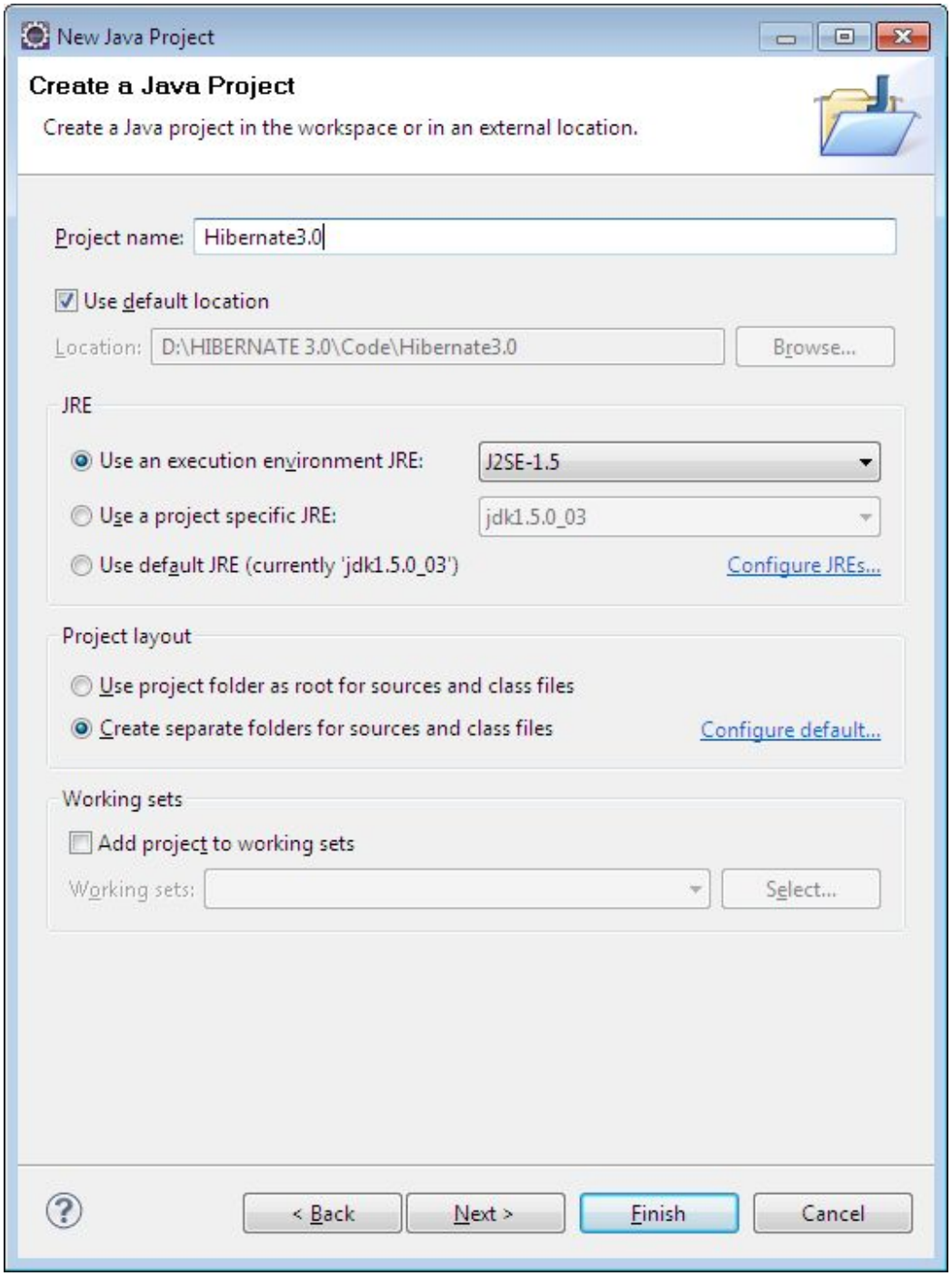
Step 2:
Type java in the wizards

adith.naveen@gmail.com



Step 3:
Give the Name as Hibernate3.0

adith.naveen@gmail.com



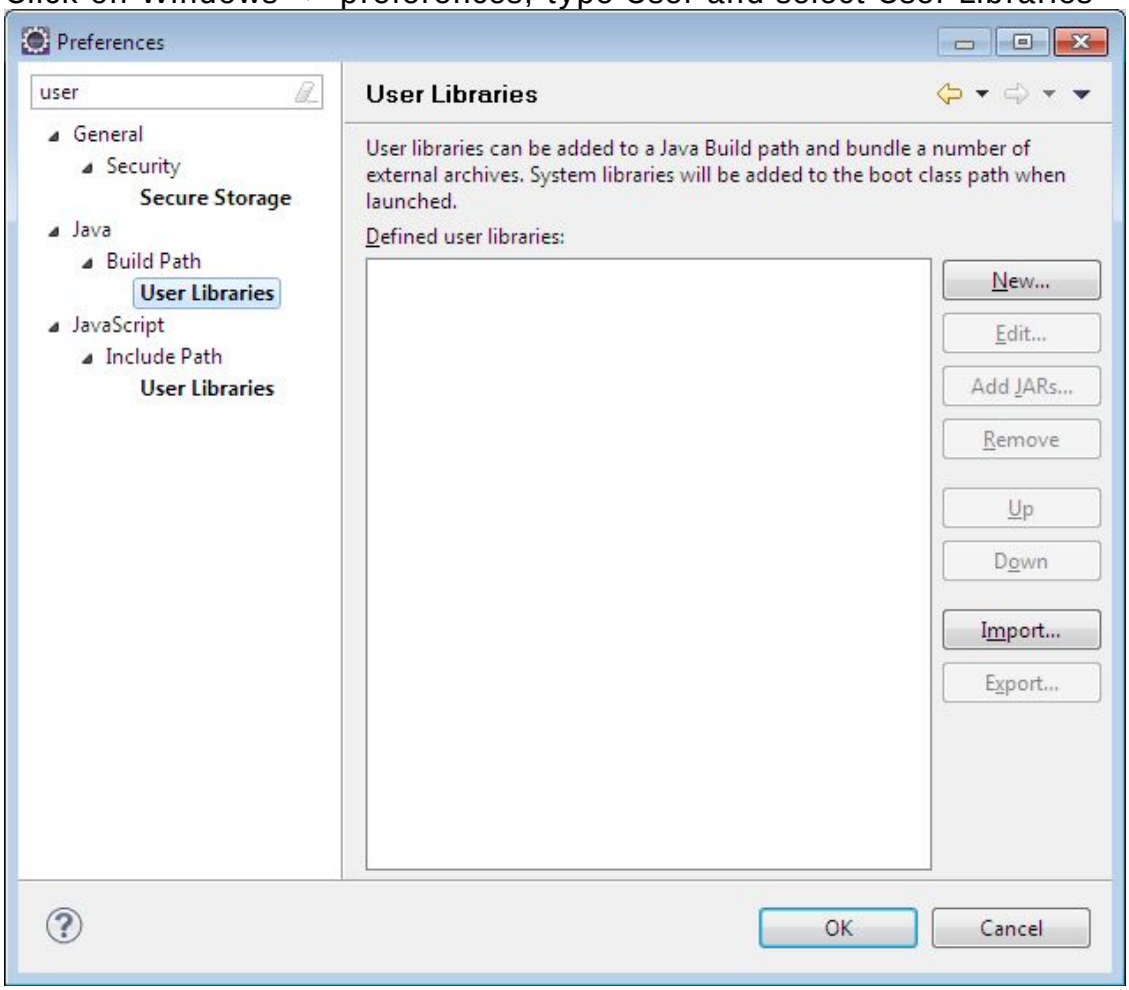
And click finish

For installation unzip the downloaded files.
Now configure Eclipse with Hibernate

adith.naveen@gmail.com

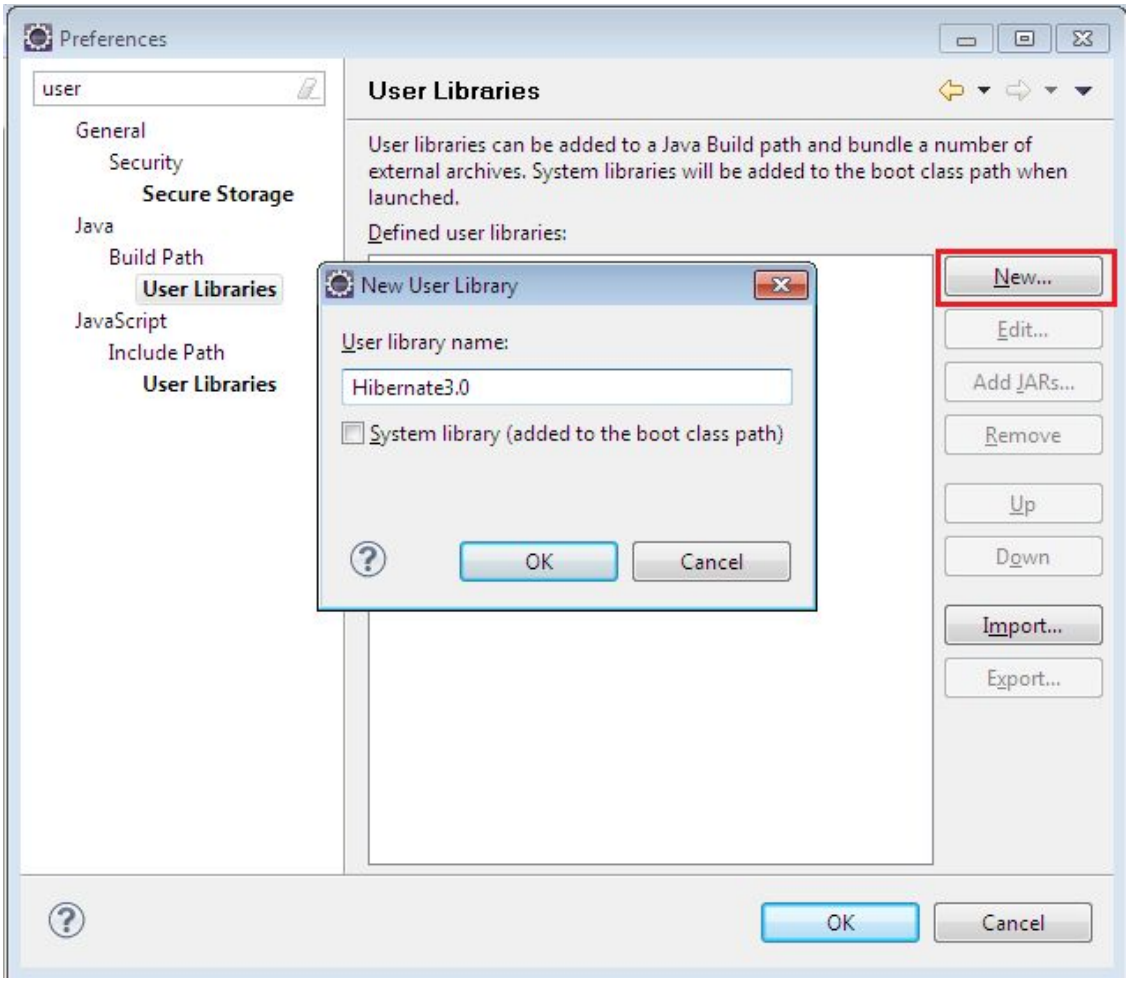
Create the user libraries to the project

Click on Windows -> preferences, type User and select User Libraries



Create a new User Library

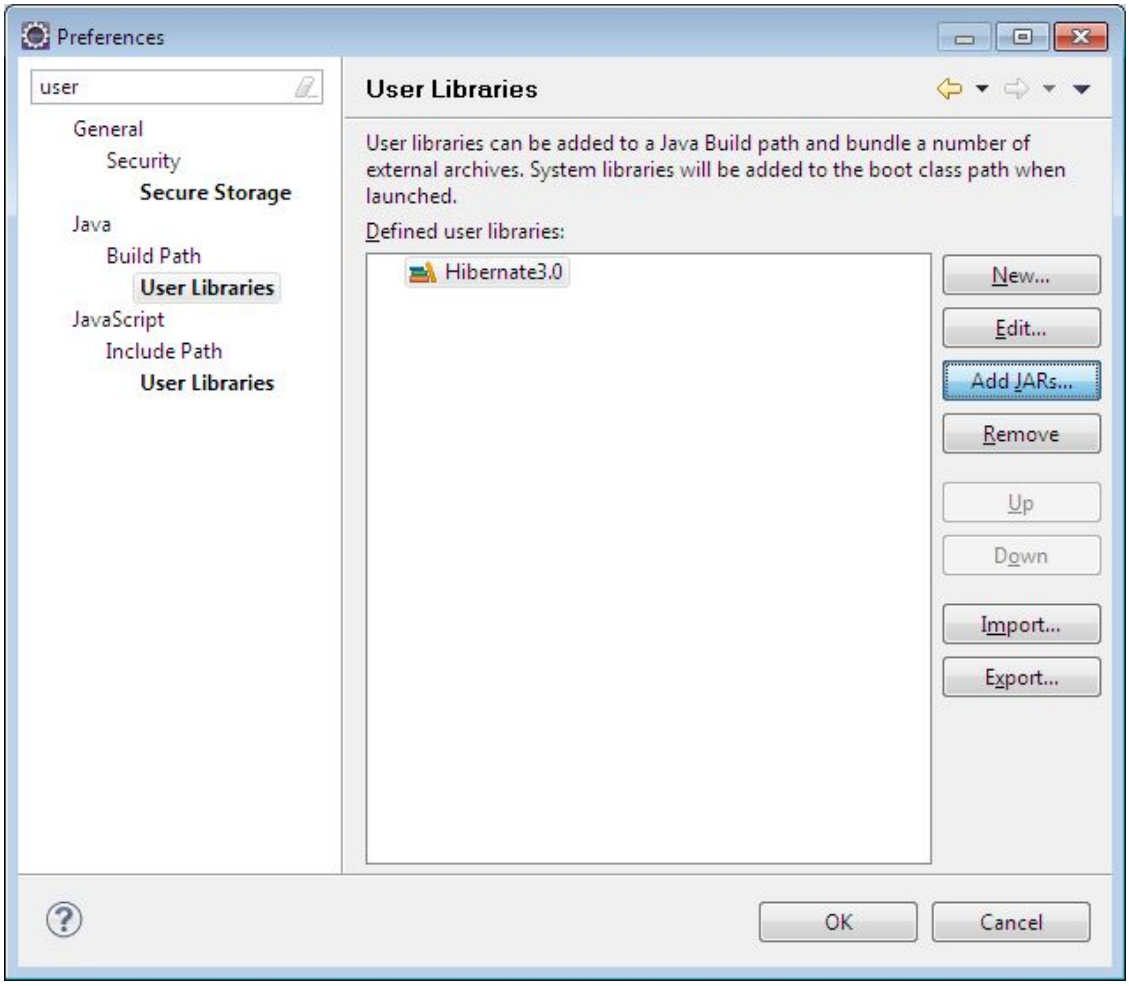
adith.naveen@gmail.com



Give Name of Library for Example (Hibernate 3.0) and say OK

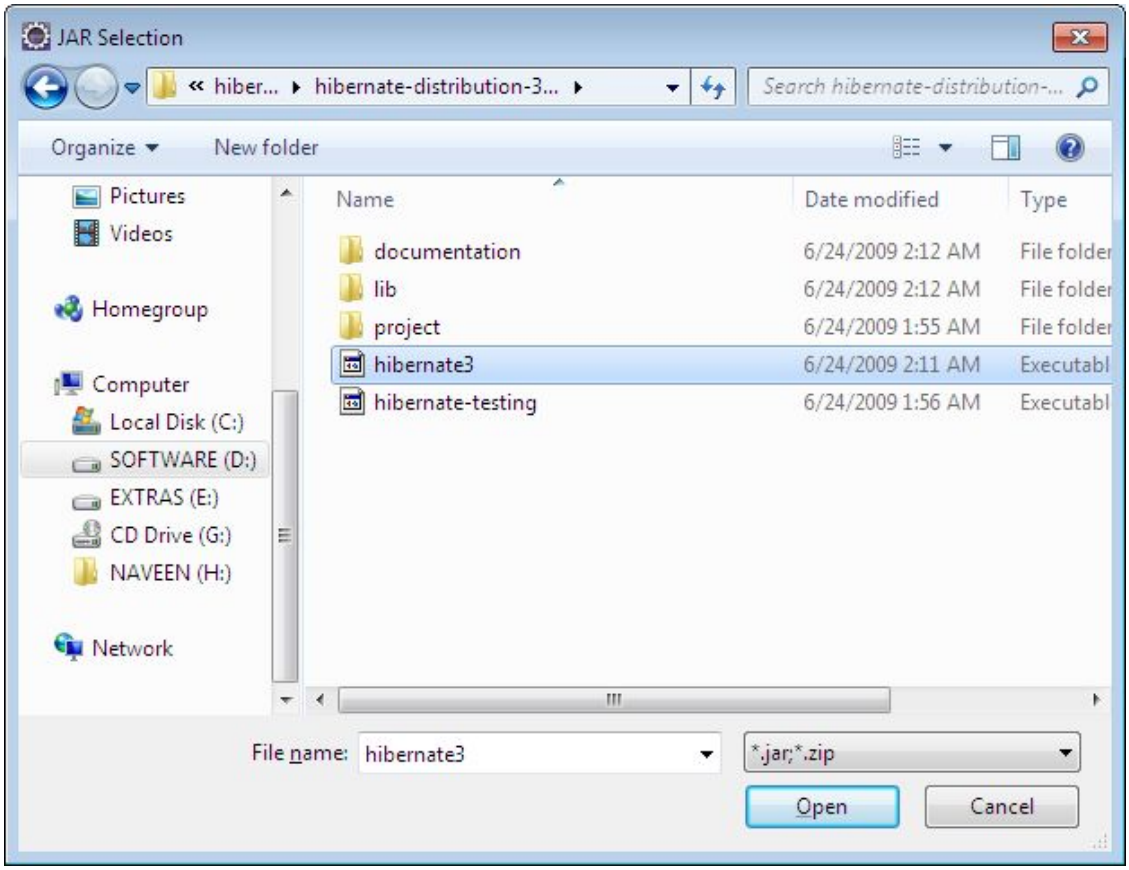
In that library add all the jar files. (Make sure you select Hibernate 3.0)

adith.naveen@gmail.com



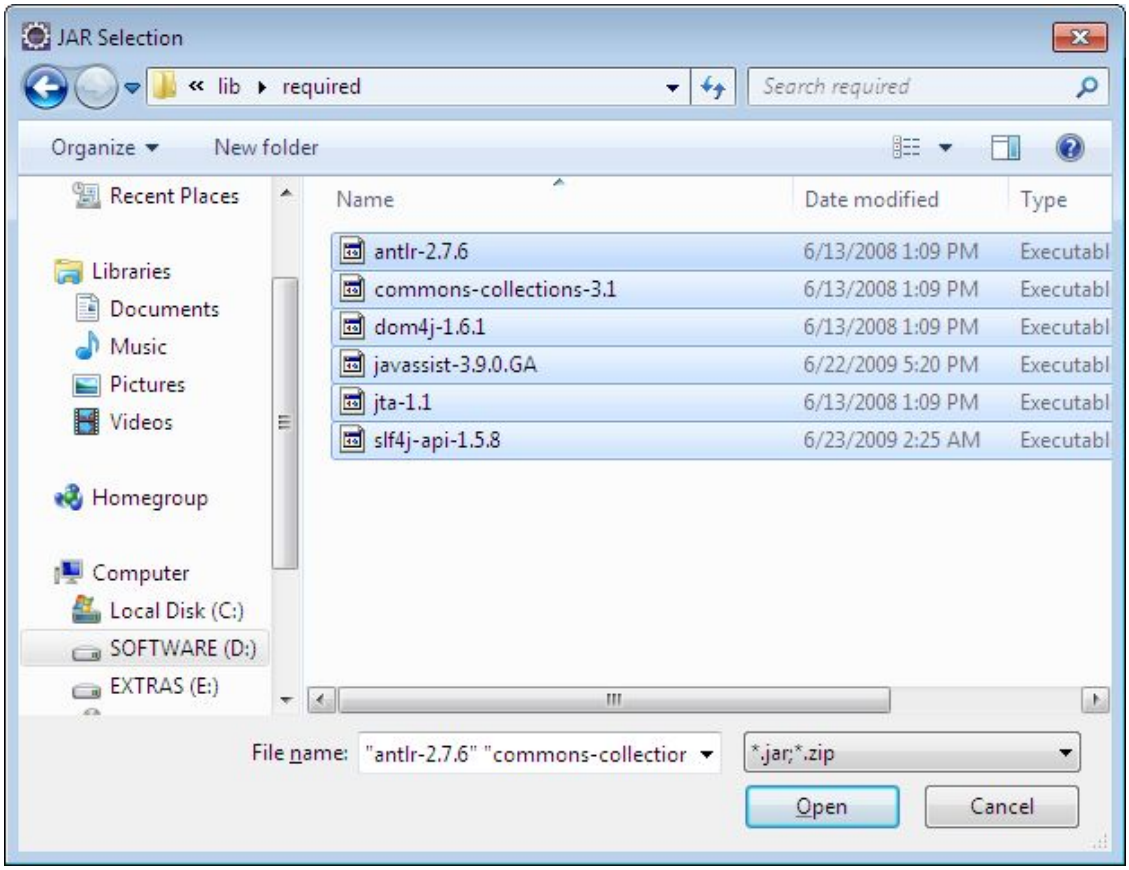
Go to Hibernate distribution folder and select hibernate3.0.jar,

adith.naveen@gmail.com



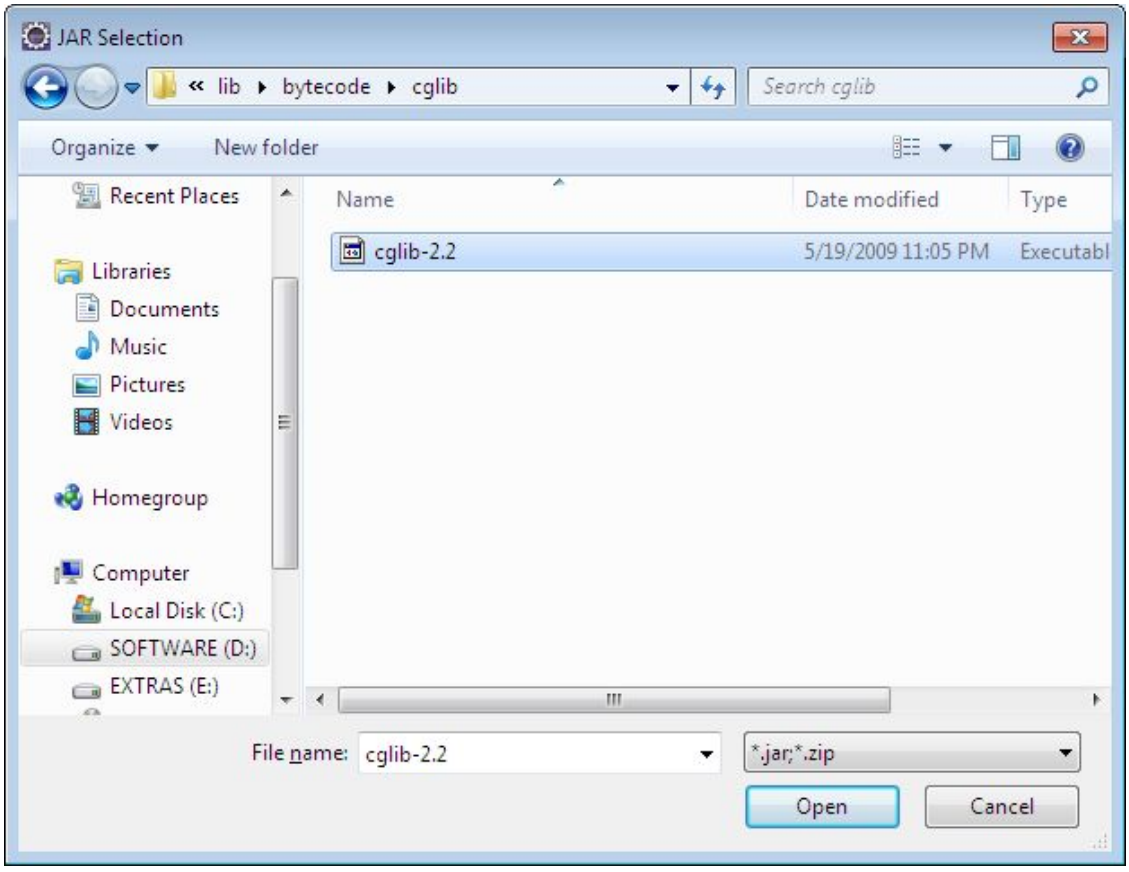
Select in lib->required folder select select all the files,

adith.naveen@gmail.com



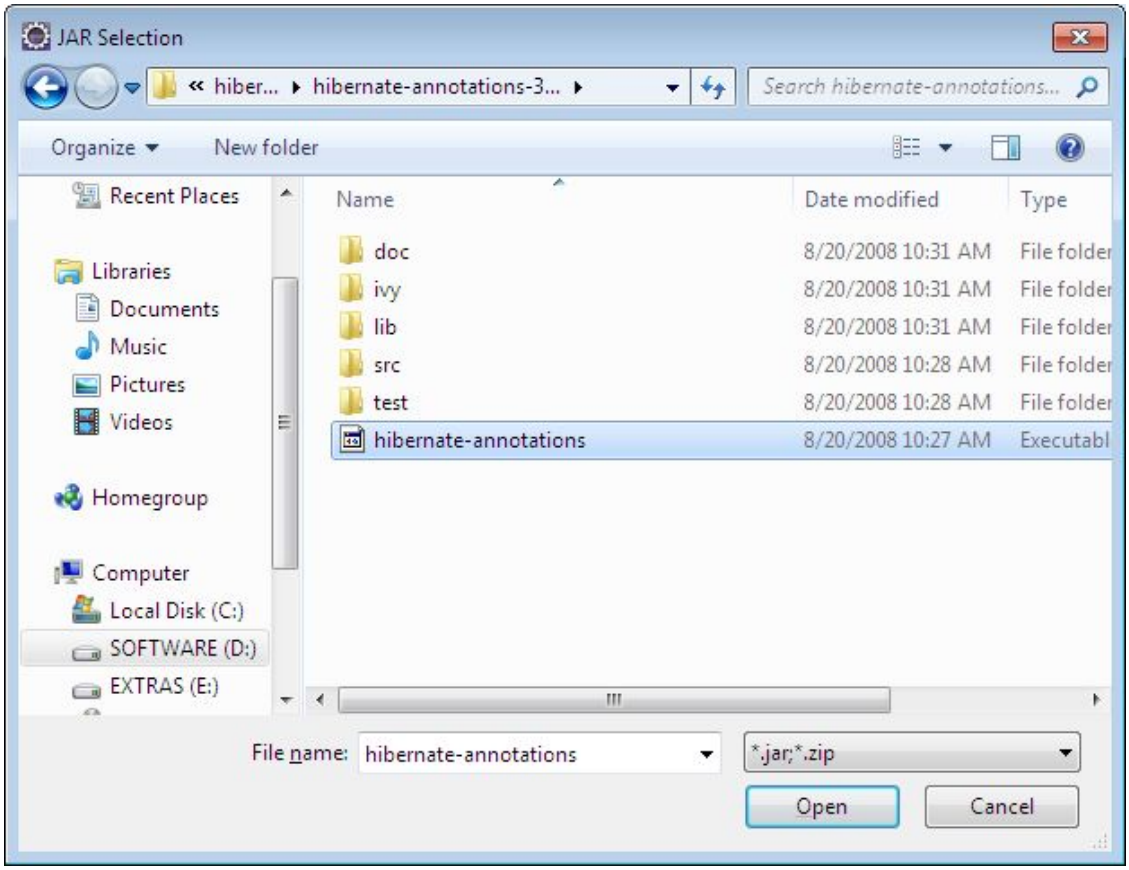
Again go to lib -> bytecode -> cglib -> cglib-2.2.jar

adith.naveen@gmail.com



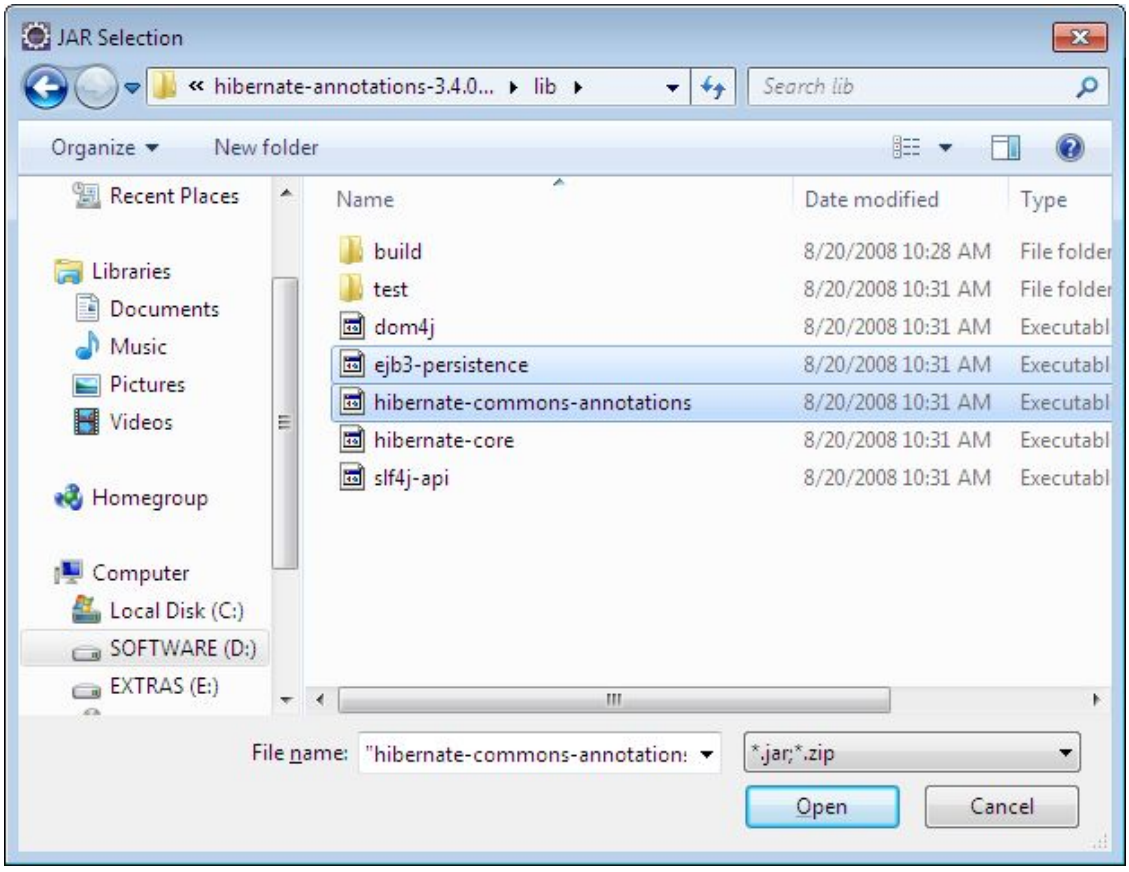
Select Hibernate Annotations -> hibernate-annotations.jar

adith.naveen@gmail.com



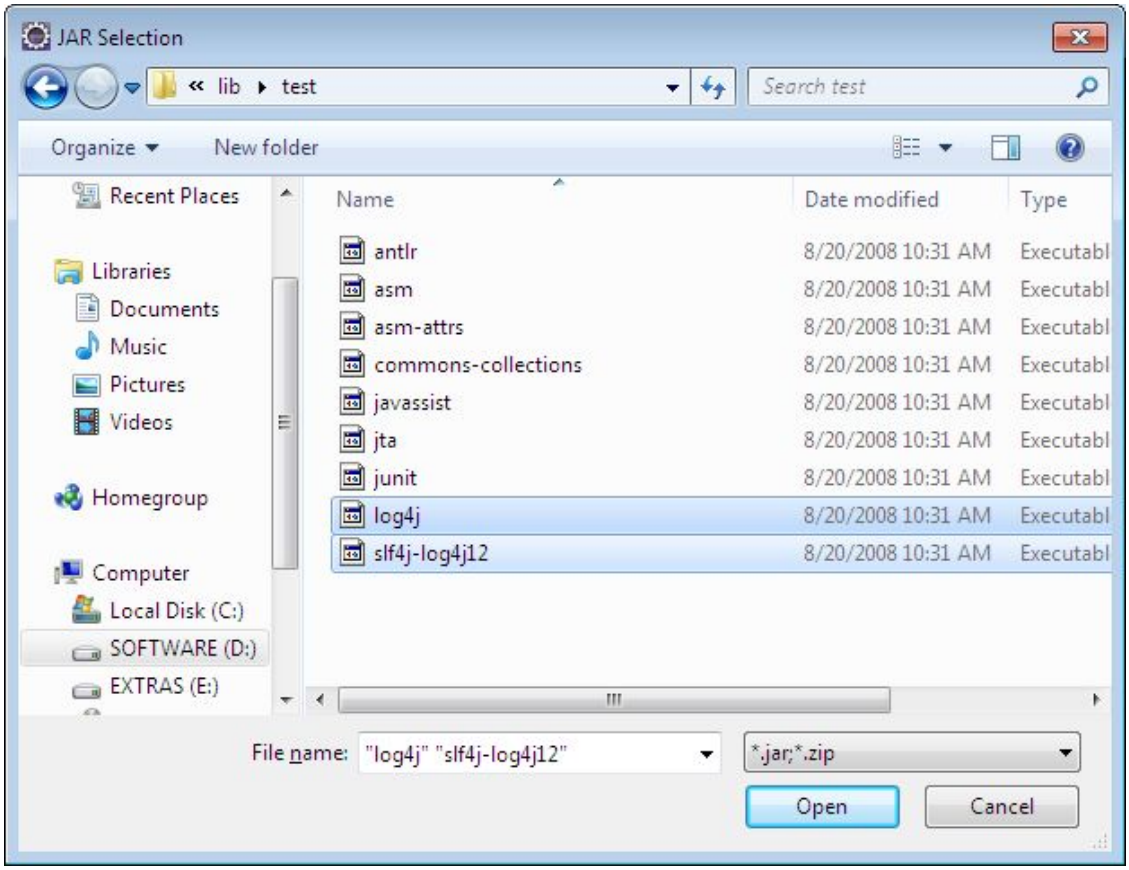
Hibernate Annotations -> lib -> ejb3-persistence.jar and hibernate-commnons-annotations.jar

adith.naveen@gmail.com

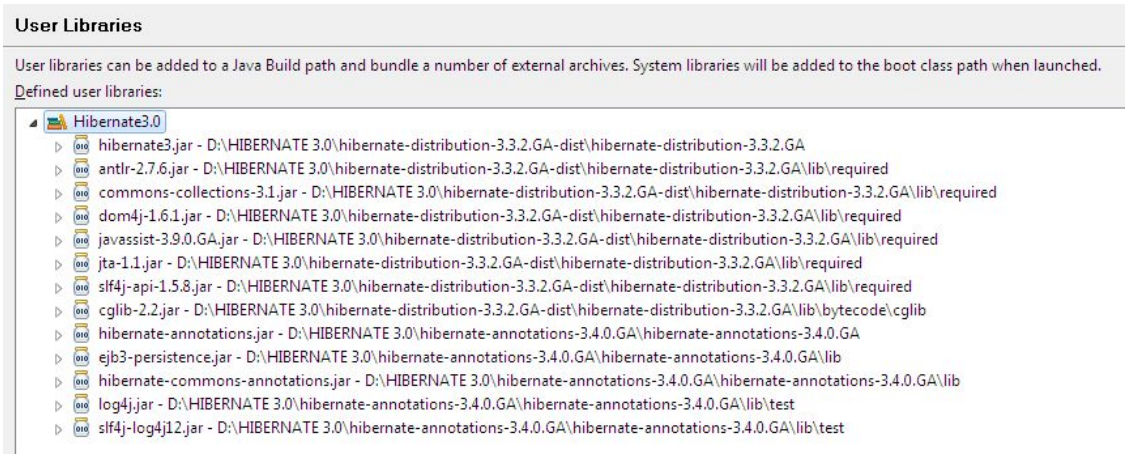


Hibernate Annotations -> lib -> test -> slf4j.log4j12.jar and log4j.jar

adith.naveen@gmail.com



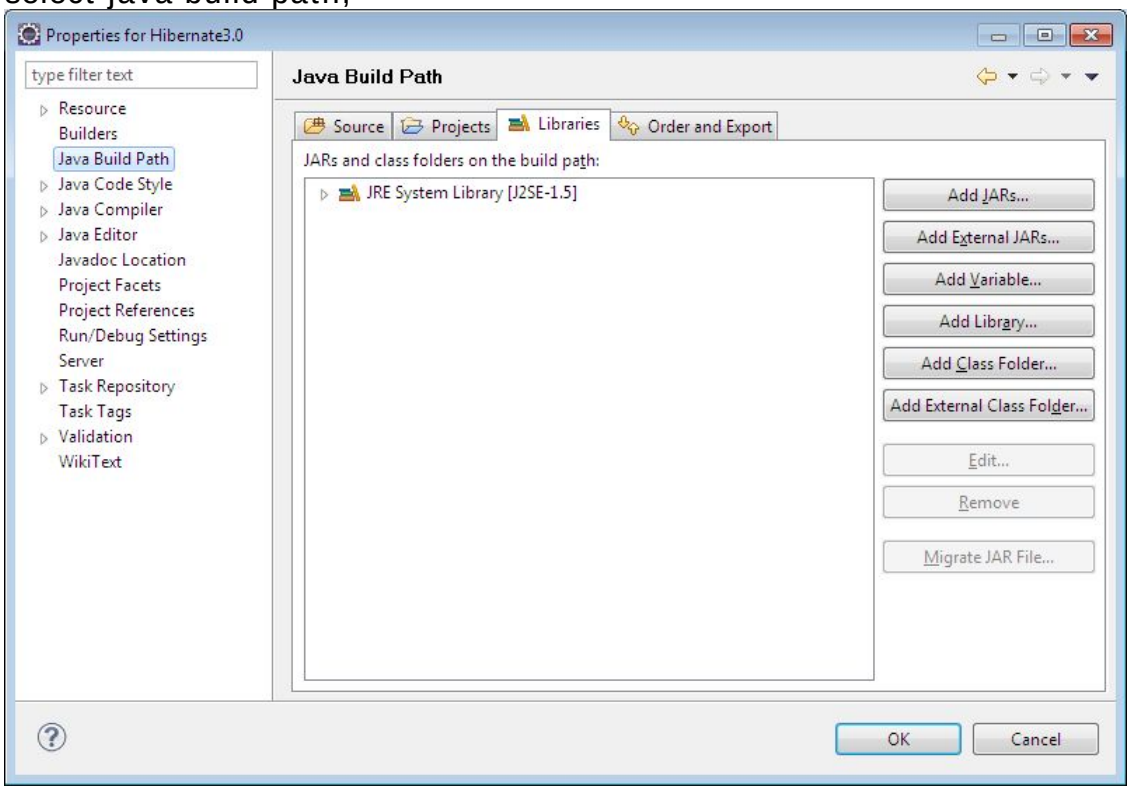
Totally there will be 13 jar files



You have created the user library, now integrate user library with Eclipse

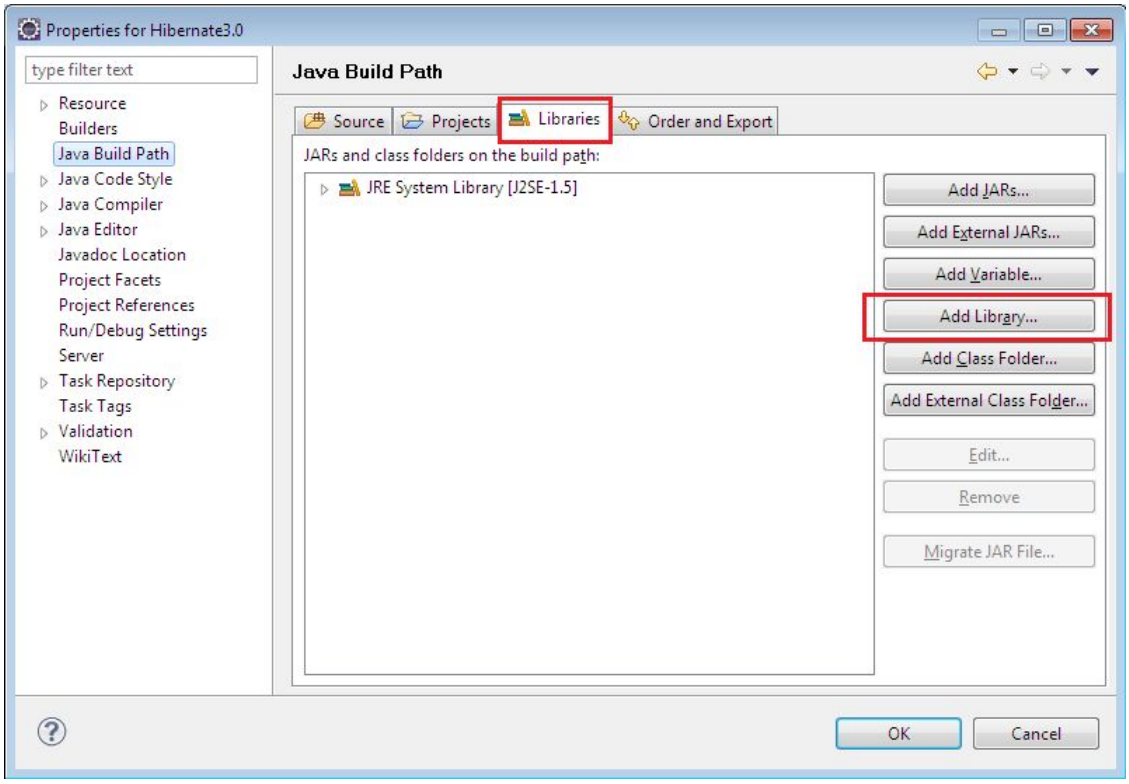
 adith.naveen@gmail.com

Select JEE or JAVA perspective, on the project click on properties, select java build path,



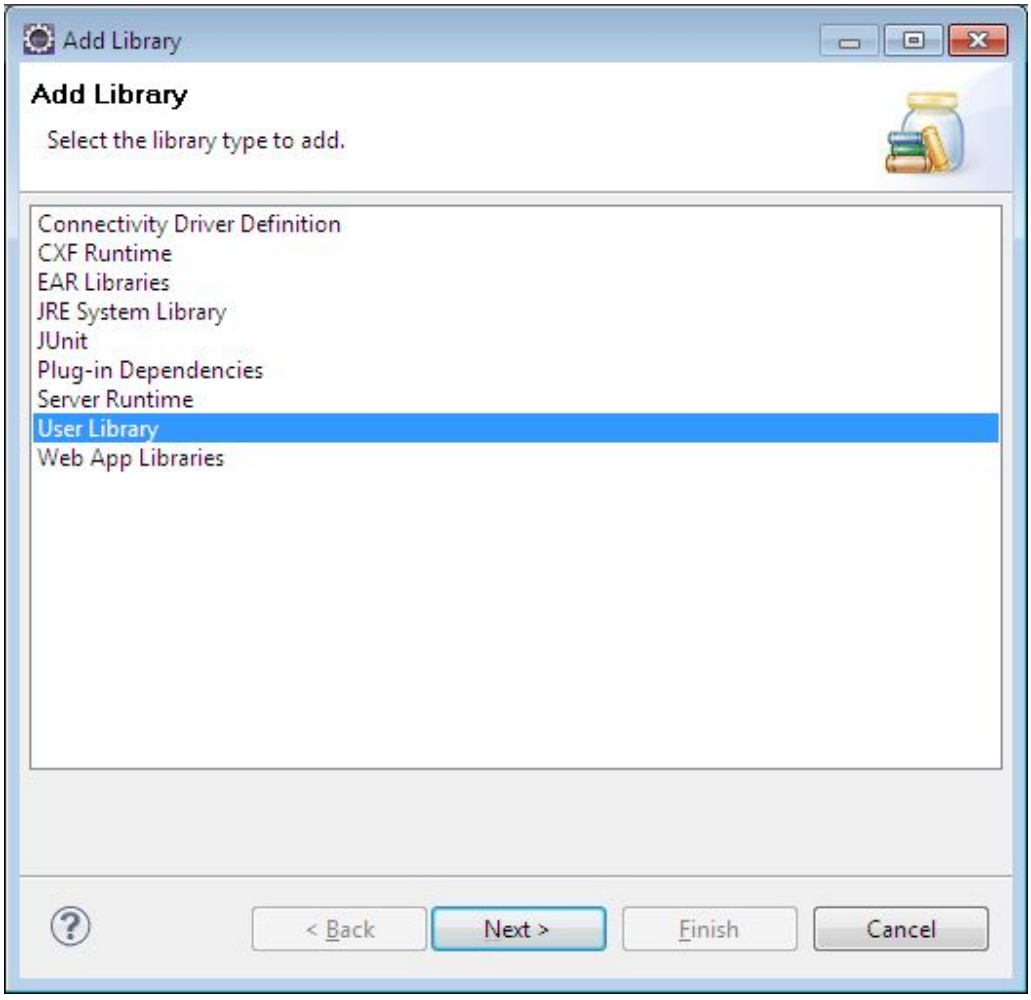
Under that select libraries, click on Add Library (on right)

adith.naveen@gmail.com



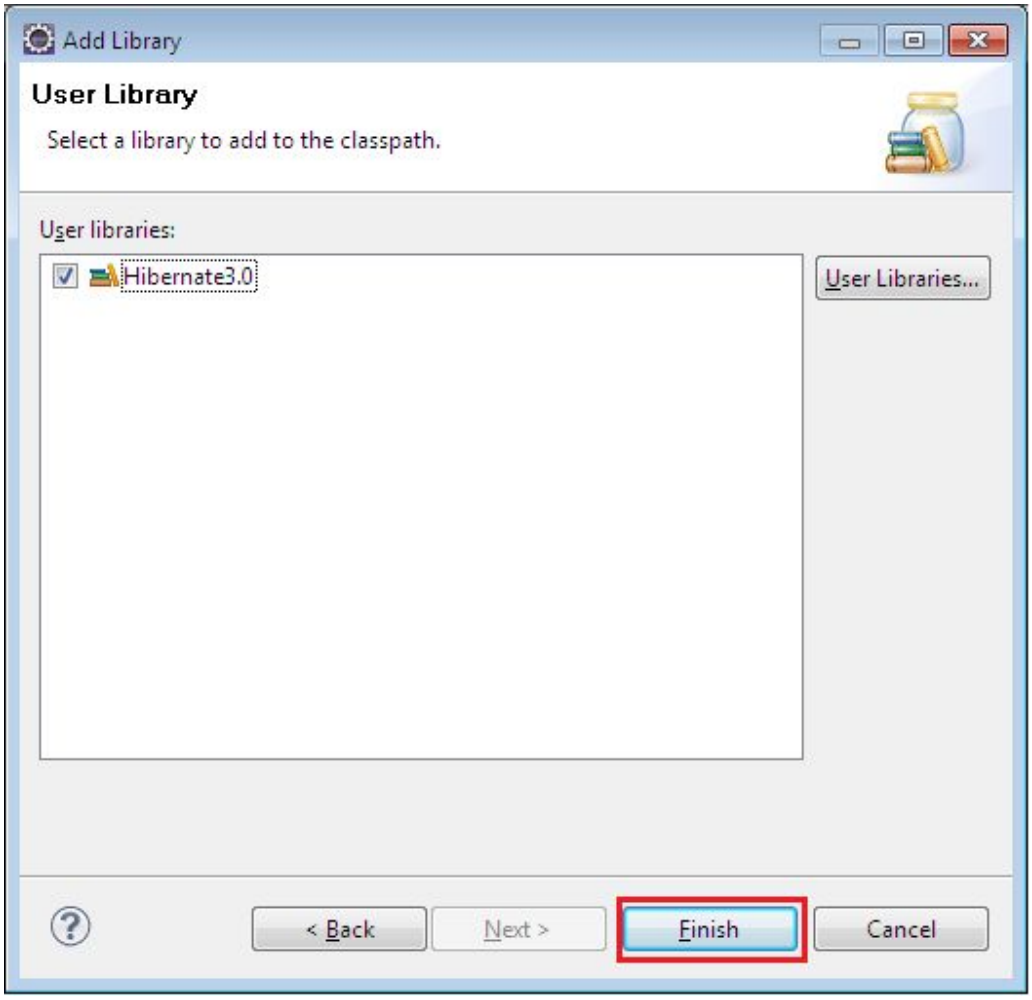
Select User Library & click next

adith.naveen@gmail.com

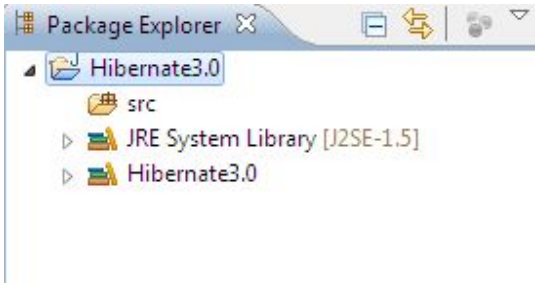


select Hibernate3.0(which is created)

adith.naveen@gmail.com



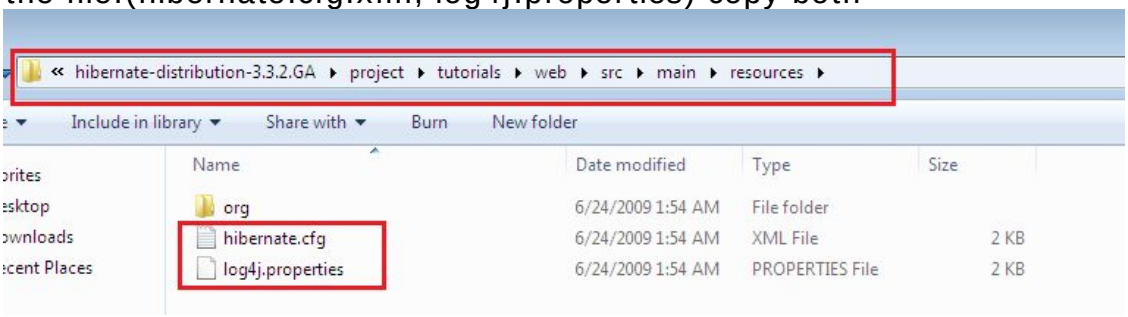
Now you should see Hibernate Library into your project.



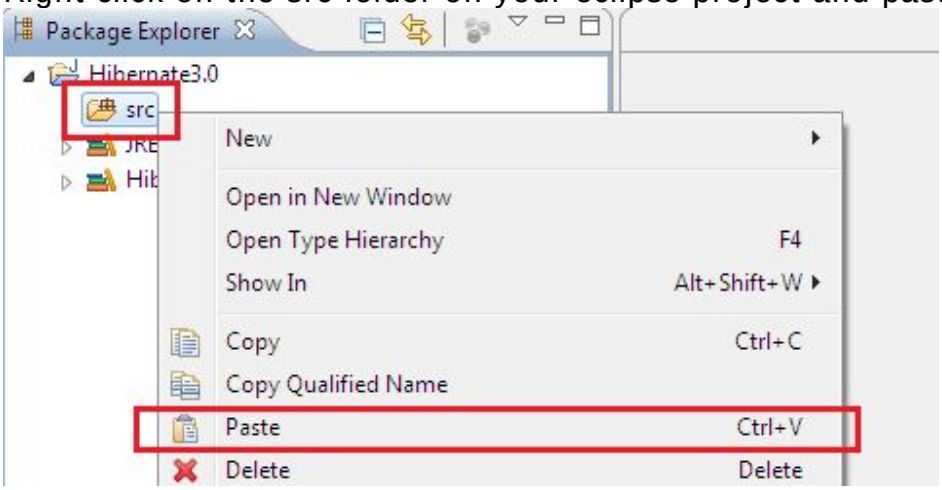
Next you have to configuration with Database and Hibernate

We have to use hibernate configuration file we need (hibernate.cfg.xml) file, this will help us to make the connection between Hibernate and Database

We can copy from the existing downloads of Hibernate Distribution, goto Project, select some tutorials, src, main, resources, and you find the file.(hibernate.cfg.xml, log4j.properties) copy both



Right click on the src folder on your eclipse project and paste on that.



Now open hibernate.cfg.xml, and now we have to configure, goto source

In the file find for the property tag with “connection .driver_class” and chage the Class name which you have, and same for URL, UserName, Pwd

```
<!-- Database connection settings -->
<property name="connection.driver_class">sun.jdbc.odbc.JdbcOdbcDriver</property>
<property name="connection.url">jdbc:odbc:HiDSN</property>
<property name="connection.username">root</property>
<property name="connection.password">naveen</property>
```

 adith.naveen@gmail.com

To find the dialect -> goto userlibrary, Hibernate3.0, and you see that it will be some where in org.hibernate.(some thing), in hibernate3.jar you will find the dialect matching your Database, and type the same in the file.

And make an entry in the configuration file

In current_session_context_class as thread
And rest keep it as it is

Comment the code property hbm2ddl.auto (because we will create the database)
And mapping resources as well since we will use our own mappings (annotations).

Note: To comment select the text and go to source and add Block Comment or ctrl+shift+/

 adith.naveen@gmail.com

```
        <!-- Database connection settings -->
        <property name="connection.driver_class">sun.jdbc.odbc.JdbcOdbcDriver</property>
        <property name="connection.url">jdbc:odbc:HiDSN</property>
        <property name="connection.username">root</property>
        <property name="connection.password">naveen</property>

        <!-- JDBC connection pool (use the built-in) -->
        <property name="connection.pool_size">2</property>

        <!-- SQL dialect -->
        <property name="dialect">org.hibernate.dialect.MySQLDialect</property>

        <!-- Enable Hibernate's current session context -->
        <property name="current_session_context_class">thread</property>

        <!-- Disable the second-level cache  -->
        <property name="cache.provider_class">org.hibernate.cache.NoCacheProvider</property>

        <!-- Echo all executed SQL to stdout -->
        <property name="show_sql">true</property>

        <!-- Drop and re-create the database schema on startup --><!--
        <property name="hbm2ddl.auto">create</property>

        <mapping resource="org/hibernate/tutorial/domain/Event.hbm.xml"/>
        <mapping resource="org/hibernate/tutorial/domain/Person.hbm.xml"/>

    --></session-factory>
```

To record the log

End of configuration.

adith.naveen@gmail.com