

[Home](#)[Java ▾](#)[JavaEE ▾](#)[Frameworks ▾](#)[Spring ▾](#)[Web Services ▾](#)[Web ▾](#)[Misc ▾](#)[Tools & Tips](#)[HOME >> HIBERNATE](#)

Hibernate Application in Eclipse IDE

July 19, 2014 by Mukesh Kumar at 3:47 pm

Basic Hibernate Application :

In this article, we are going to create a simple example of hibernate application using eclipse IDE. After finishing of this article You will learn how to create and run the Hibernate based projects in Eclipse IDE, via XML mapping file (hbm).

Tools and Technologies :

- JDK 1.6
- Hibernate 3.6.3.Final
- Eclipse
- MySQL 5.0.

For developing first hibernate application in Eclipse, we need to follow bellow steps:

Step 1 : Create the java project :

Open Eclipse IDE, click on File-> New -> project -> Java and select java project. Now specify the project name as FirstHibernateEx then click on next -> finish .

Hibernate Tutorial

[Hibernate Basics](#)[Hibernate Inheritance Mapping](#)[Hibernate Query Language \(HQL\)](#)[Composite Primary Keys In Hibernate](#)

GOODBYE.

FEEDTHEPIG.ORG



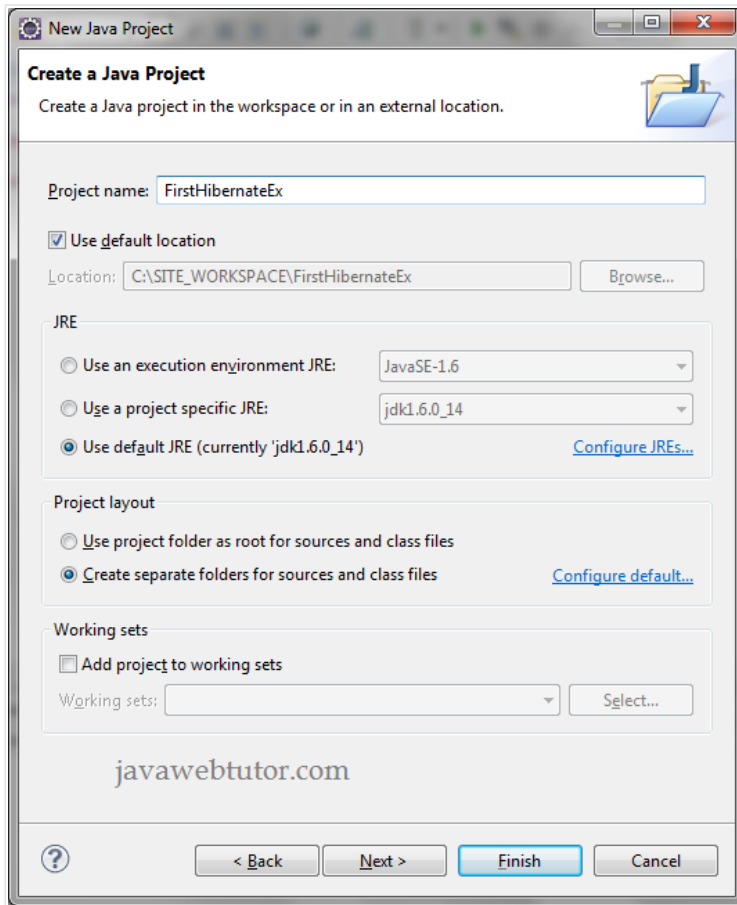
**STOP TEXTS
STOP WRECKS**

LEARN MORE



Popular Posts

- [CRUD Example using Spring MVC, Hibernate, Maven and MySQL](#)
- [Hibernate Query Language HQL select, Update And Delete Example](#)
- [Hibernate Query Language](#)
- [JSF Page Navigation Tutorial](#)
- [JSF Validation Tutorial](#)
- [Hibernate One-To-Many XML Mapping](#)

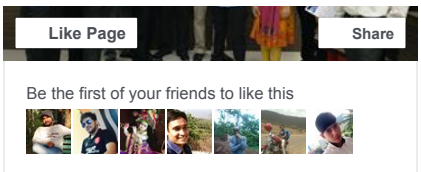


- JSF Example: Using JSF 2.0, Eclipse, And Maven
- Jboss Tools Installation in Eclipse
- Jboss Installation in Eclipse
- Database Application Using JSF
- Java Web Application With Maven
- Spring HelloWorld Example Using Eclipse And Maven
- EJB3 and JPA Example

Mukesh Kumar

Follow

640 likes

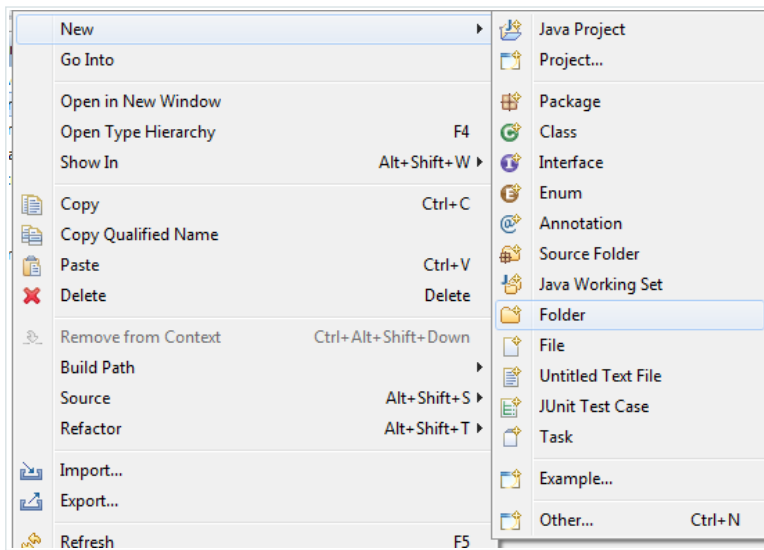


Step 2 : Add Jar files for hibernate and mysql :

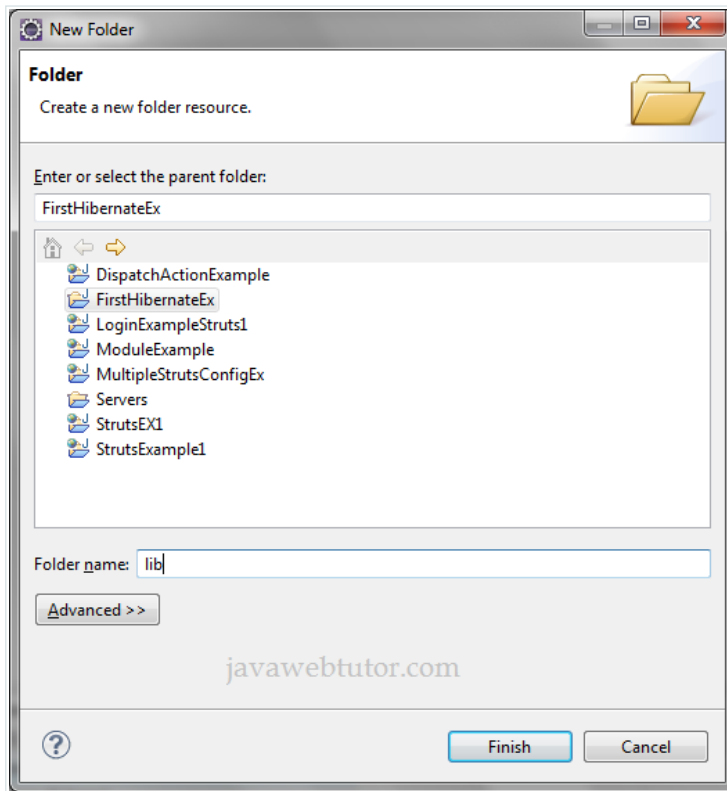
First we need to create one folder in the project, so that we can place all required jar files in one place.

Right Click on your Project and select New -> Folder as shown below





In New Folder screen provide Folder name as lib(Dont confuse with lib we can assign any name to the folder) and Click Finish as shown below .



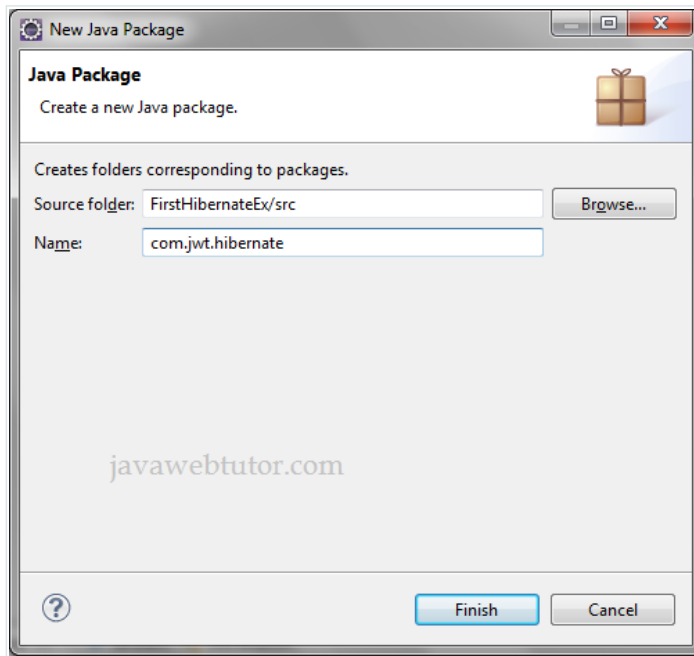
Download the latest hibernate jar files from [Here](#). After downloading the Jar files copy following Jar files inside the lib folder which we created earlier.

- antlr-2.7.6
- commons-collections-3.1
- dom4j-1.6.1
- hibernate3
- javassist-3.4.GA
- jta-1.1
- slf4j-api-1.5.6
- slf4j-simple-1.5.6
- mysql

Now add these jar files to classpath of your project.

Step 3 : Create the persistence class :

Create a package by Right Click on your Project and navigate to New -> Package and provide the name com.jwt.hibernate to the package and click on Finish as shown bellow.



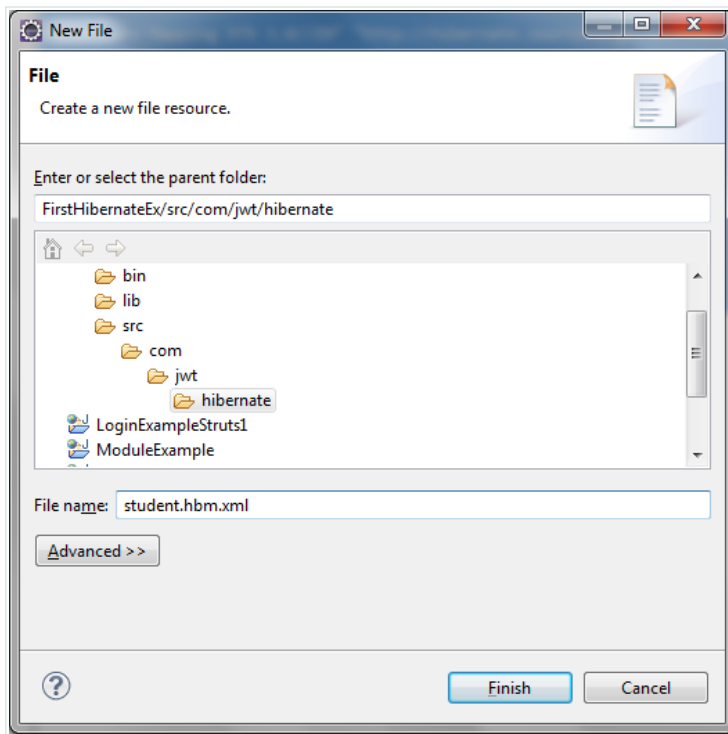
Now create a java class **Student.java** in this package by right click on the **package** -> **New** -> **Class** and add following code in this class.

Student.java

```
1  package com.jwt.hibernate;
2
3  public class Student {
4  private long id;
5  private String name;
6  private String degree;
7  private String roll;
8  private String phone;
9
10 public long getId() {
11     return id;
12 }
13 public void setId(long id) {
14     this.id = id;
15 }
16 public String getName() {
17     return name;
18 }
19 public void setName(String name) {
20     this.name = name;
21 }
22 public String getDegree() {
23     return degree;
24 }
25 public void setDegree(String degree) {
26     this.degree = degree;
27 }
28 public String getRoll() {
29     return roll;
30 }
31 public void setRoll(String roll) {
32     this.roll = roll;
33 }
34 public String getPhone() {
35     return phone;
36 }
37 public void setPhone(String phone) {
38     this.phone = phone;
39 }
40 }
```

Step 4 : Create the mapping file for Persistent class :

Right click on your package then navigate to New -> File and provide the name as student.hbm.xml and click Finish.



Add below code into student.hbm.xml file

student.hbm.xml

```

1  <?xml version="1.0" encoding="UTF-8"?>
2  <!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"
3  <hibernate-mapping>
4  <class name="com.jwt.hibernate.Student" table="STUDENT">
5  <id column="ID" name="id" type="long" />
6  <property column="STUDENT_NAME" name="name" type="string" />
7  <property column="DEGREE" name="degree" type="string" />
8  <property column="ROLL" name="roll" type="string" />
9  <property column="PHONE" name="phone" type="string" />
10 </class>
11 </hibernate-mapping>

```

In this mapping file, Student class is linked with STUDENT table in the database, and next is the id element, means in the database table what column we need to take as primary key column, that property name we need to give here, I have been given my property name id which will mapped with ID column in the table, name is mapped with STUDENT_NAME column, degree is mapped with DEGREE column of the STUDENT table and so on.

Note :- If your database column name and persistence class variable is same then no need to specify column property in .hbm file.

Step 5 : Create the Configuration file :

The configuration file contains informations about the database and mapping file. Conventionally, its name should be hibernate.cfg.xml. Configuration file must be in classpath of your Project. Place this file in src of your project by default it will added to classpath of your project.

hibernate.cfg.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <!DOCTYPE hibernate-configuration PUBLIC
3  "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
4  "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
5
6  <hibernate-configuration>
7  <session-factory>
8  <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>
9  <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/javawebtutor</property>
10 <property name="hibernate.connection.username">root</property>
11 <property name="hibernate.connection.password">mukesh</property>
12 <property name="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

```

```

13 <property name="show_sql">true</property>
14 <property name="format_sql">true</property>
15 <property name="hbm2ddl.auto">create </property>
16 <mapping resource="com/jwt/hibernate/student.hbm.xml" />
17 </session-factory>
18 </hibernate-configuration>

```

Note :- In the above configuration file we mentioned `<property name="hbm2ddl.auto">create</property>`. If we are assigning this value as create, it will create table in database automatically when we run the application. So no need to write sql query for table creation. But when you are running the example second times you need to change this attribute to "update" otherwise hibernate will throw exception.

Step 6: Create the Test class that stores the object :

In this class, we are simply storing the student object to the database.

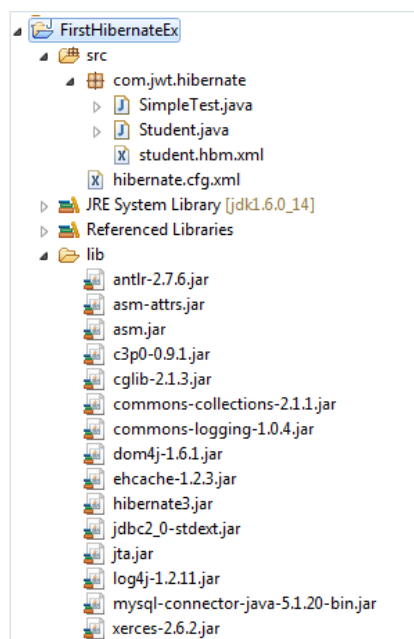
SimpleTest.java

```

1  package com.jwt.hibernate;
2
3  import org.hibernate.Session;
4  import org.hibernate.SessionFactory;
5  import org.hibernate.Transaction;
6  import org.hibernate.cfg.Configuration;
7
8  public class SimpleTest {
9
10     public static void main(String[] args) {
11
12         Configuration cfg = new Configuration();
13         cfg.configure("hibernate.cfg.xml");
14
15         SessionFactory factory = cfg.buildSessionFactory();
16         Session session = factory.openSession();
17         Student student = new Student();
18         student.setName("Mukesh");
19         student.setRoll("101");
20         student.setPhone("8888");
21         student.setDegree("B.E");
22
23         Transaction tx = session.beginTransaction();
24         session.save(student);
25         System.out.println("Object saved successfully.....!!");
26         tx.commit();
27         session.close();
28         factory.close();
29     }
30 }

```

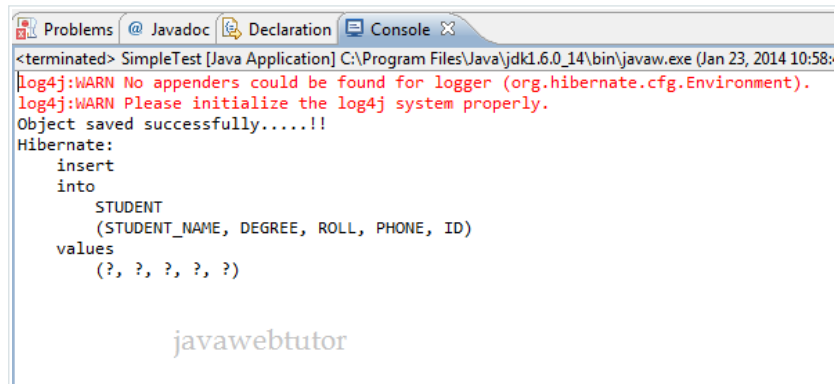
Directory structure of the project :



Run the Application :

To run the hibernate application, right click on the SimpleTest class - Run As - Java Application.

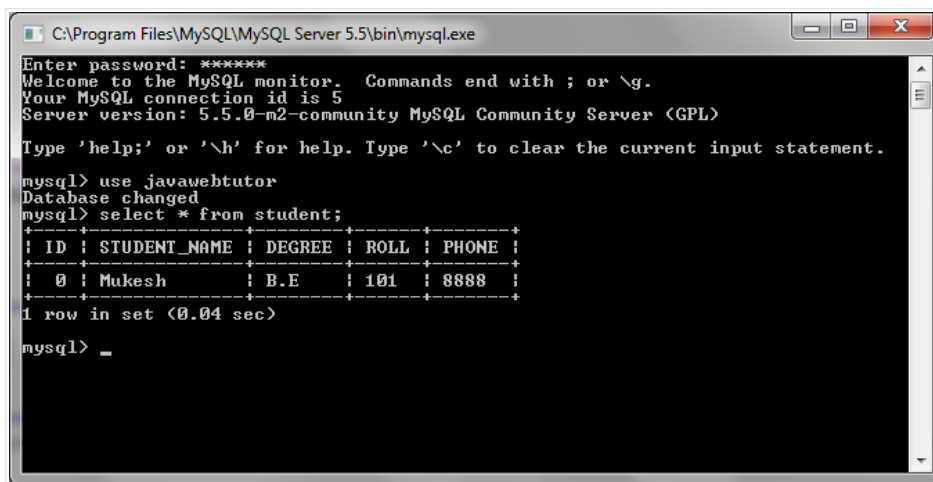
Output in Eclipse :



```
<terminated> SimpleTest [Java Application] C:\Program Files\Java\jdk1.6.0_14\bin\javaw.exe (Jan 23, 2014 10:58:
log4j:WARN No appenders could be found for logger (org.hibernate.cfg.Environment).
log4j:WARN Please initialize the log4j system properly.
Object saved successfully.....!!
Hibernate:
insert
into
STUDENT
(STUDENT_NAME, DEGREE, ROLL, PHONE, ID)
values
(?, ?, ?, ?, ?)
```

javawebtutor

In The Database :



```
C:\Program Files\MySQL\MySQL Server 5.5\bin>mysql.exe
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.5.0-m2-community MySQL Community Server (GPL)
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use javawebtutor
Database changed
mysql> select * from student;
+----+-----+-----+-----+-----+
| ID | STUDENT_NAME | DEGREE | ROLL | PHONE |
+----+-----+-----+-----+-----+
| 0  | Mukesh      | B.E   | 101  | 8888  |
+----+-----+-----+-----+-----+
1 row in set (0.04 sec)

mysql> _
```

You can download the source code of the example by clicking on the Download link below.

Source + Lib : [Download](#)

Related Articles

1. Web Application with Hibernate
2. CRUD Operations Using Hibernate 3
3. Inheritance Mapping In Hibernate
4. Hibernate Example With Annotation



[Index](#)



Related Ads

[Install Java Script](#)[Java Web](#)[java JFrame](#)[Java Flash Player](#)[Computer Programming](#)[Javascript Fixer Download](#)2 Comments [javawebtutor](#)[Login](#)[Recommend](#) [Share](#)[Sort by Best](#)

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS [?](#)

Name

**Anup Sonkusare** • 4 months ago

Thanks a lot...!! I never seen such a beautiful explanation of hibernate technology anywhere.. You explained it in very simple and lucid language. Thank you very much again..and keep posting such wonderful stuff..!

[^](#) | [v](#) • [Reply](#) • [Share](#)**Joseph Mtinangi** • 8 months ago

This is very helpful. Thank you very much.

[^](#) | [v](#) • [Reply](#) • [Share](#)

ALSO ON JAVAWEBTUTOR

Accessing Databases from JSP Pages

2 comments • a year ago •



ccleaner — ::after </div> <script> alert("is this working?");</script>

Web Application With Hibernate,JSP and Servlet using Eclipse

13 comments • a year ago •



Godfrey J Lawrence — CREATE TABLE USER (id INT NOT NULL AUTO_INCREMENT, userName VARCHAR(100) NOT NULL, password1 ...

[Top of Page](#)**Web Application With Hibernate using Eclipse**

3 comments • a year ago •



Manali Patel — i searched alot..finally this was a relief,,thank you..instead of using servlet is there any options of using jsf xhtml files sir..?

Hibernate One-to-Many XML Mapping Tutorial

1 comment • a year ago •



Eduardo Blanco — Hi, bice job with this tutorial, it is very well explained and works very fine, but I have a question how can I do to insert a record ...

[About Us](#)[Privacy Policy](#)[Contact Us](#)

javawebtutor.com © 2013-2016

DISCLAIMER

Oracle & Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners. javawebtutor.com is not connected to Oracle Corporation and is not sponsored by Oracle Corporation. The Examples &

ABOUT US

javawebtutor.com is a site dedicated to bringing you the coolest java and related web development tutorials and resources.

[ABOUT US](#)**FREE EBOOKS**

Download ebooks.

[DOWNLOAD IT](#)

Tutorial provided here are for learning purpose only. we do not warrant the correctness of its content. The risk from using it lies entirely with the user.

[LEARN MORE](#)