**Source Code for Adding a New Product in the Database.**

1. Create table In database “userdetails”
2. Create dynamic folder
3. Add jar files
4. Create index.html file
5. Create hibernate.cfg.xml file
6. Create userdetails.hbm.xml file
7. Create hibernatetestservlet servlet file
8. Create class od userdetails

4) Create index.html file

<!DOCTYPEhtml>

<html>

<head>

<metacharset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<formaction=*"HibernateTestServlet"*method=*"get"*>

Enter userId<inputtype=*"text"*name=*"txtUid"*><br><br>

Enter Name<inputtype=*"text"*name=*"txtName"*><br><br>

<inputtype=*"submit"*value=*"save"*>

</body>

</html>

5) Create hibernate.cfg.xml file

<?xmlversion=*"1.0"*encoding=*"UTF-8"*?>

<!--

~ Hibernate, Relational Persistence for Idiomatic Java

~

~ License: GNU Lesser General Public License (LGPL), version 2.1 or later.

~ See the lgpl.txt file in the root directory or <http://www.gnu.org/licenses/lgpl-2.1.html>.

-->

<!DOCTYPEhibernate-configurationPUBLIC

"-//Hibernate/Hibernate Configuration DTD//EN"

"http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<propertyname=*"format\_sql"*>false</property>

<propertyname=*"connection.url"*>jdbc:mysql://localhost:3306/My\_database</property>

<propertyname=*"connection.driver\_class"*>com.mysql.cj.jdbc.Driver</property>

<propertyname=*"connection.username"*>root</property>

<propertyname=*"connection.password"*>Yogesh@01</property>

<!-- JDBC connection pool (use the built-in) -->

<propertyname=*"connection.pool\_size"*>1</property>

<!-- SQL dialect -->

<propertyname=*"dialect"*>org.hibernate.dialect.MySQL5Dialect</property>

<!-- Disable the second-level cache -->

<propertyname=*"cache.provider\_class"*>org.hibernate.cache.internal.NoCacheProvider</property>

<!-- Echo all executed SQL to stdout -->

<propertyname=*"show\_sql"*>true</property>

<!-- Drop and re-create the database schema on startup -->

<propertyname=*"hbm2ddl.auto"*>create-drop</property>

<!--<property name="hbm2ddl.auto">update</property> -->

<mappingresource=*"UserDetails.hbm.xml"*/>

</session-factory>

</hibernate-configuration>

1. Create userdetails.hbm.xml file

<?xmlversion=*"1.0"*encoding=*"UTF-8"*?>

<!DOCTYPEhibernate-mappingPUBLIC"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<hibernate-mappingpackage=*"beans"*default-access=*"field"*>

<classname=*"UserDetails"*table=*"UserDetails"*>

<idname=*"id"*column=*"id"*type=*"java.lang.Integer"*>

</id>

<propertyname=*"username"*length=*"20"*></property>

</class>

</hibernate-mapping>

1. Create hibernatetestservlet servlet file

**package** services;

**import**java.io.File;

**import**java.io.IOException;

**import**javax.servlet.ServletException;

**import**javax.servlet.annotation.WebServlet;

**import**javax.servlet.http.HttpServlet;

**import**javax.servlet.http.HttpServletRequest;

**import**javax.servlet.http.HttpServletResponse;

**import**org.hibernate.Session;

**import**org.hibernate.SessionFactory;

**import**org.hibernate.cfg.Configuration;

**import**beans.UserDetails;

@WebServlet("/HibernateTestServlet")

**publicclass**HibernateTestServlet**extends**HttpServlet {

**privatestaticfinallong*serialVersionUID*** = 1L;

**void**register(UserDetailsuser){

SessionFactorysessionFactory=**new**Configuration().configure(**new** File("C:\\Users\\sai\\eclipse-workspace\\HybernetExample1\\src\\main\\java\\hibernate.cfg.xml")).buildSessionFactory();

Session session=sessionFactory.openSession();

session.beginTransaction();

session.save(user);

session.getTransaction().commit();

}

**protectedvoid**doGet(HttpServletRequestrequest, HttpServletResponseresponse) **throws**ServletException, IOException {

// **TODO** Auto-generated method stub

**int**uid=Integer.*parseInt*(request.getParameter("txtUid"));

String uname=request.getParameter("txtName");

//create Object of UserDetails

UserDetailsuser=**new**UserDetails();

user.setId(uid);

user.setUsername(uname);

//call register method

register(user);

}

}

1. Create class od userdetails

**package** beans;

**publicclass**UserDetails {

**privateint**id;

**private** String username;

**publicint**getId() {

**return**id;

}

**publicvoid**setId(**int**id) {

**this**.id = id;

}

**public** String getUsername() {

**return**username;

}

**publicvoid**setUsername(String username) {

**this**.username = username;

}

}