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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*"HibernateTestServlet"* method=*"get"*>

Enter userId<input type=*"text"* name=*"txtUid"*><br><br>

Enter Name<input type=*"text"* name=*"txtName"*><br><br>

<input type=*"submit"* value=*"save"*>

</body>

</html>

5) Create hibernate.cfg.xml file

<?xml version=*"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>.

-->

<!DOCTYPE hibernate-configuration PUBLIC

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

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

<hibernate-configuration>

<session-factory>

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

<property name=*"connection.url"*>jdbc:mysql://localhost:3306/Mphasis</property>

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

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

<property name=*"connection.password"*>Iqbal</property>

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

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

<!-- SQL dialect -->

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

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

<property name=*"cache.provider\_class"*>org.hibernate.cache.internal.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-drop</property>

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

<mapping resource=*"UserDetails.hbm.xml"*/>

</session-factory>

</hibernate-configuration>

1. Create userdetails.hbm.xml file

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN"

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

<hibernate-mapping package=*"beans"* default-access=*"field"*>

<class name=*"UserDetails"* table=*"UserDetails"* >

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

</id>

<property name=*"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")

**public** **class** HibernateTestServlet **extends** HttpServlet {

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

**void** register(UserDetails user){

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

Session session=sessionFactory.openSession();

session.beginTransaction();

session.save(user);

session.getTransaction().commit();

}

**protected** **void** doGet(HttpServletRequest request, HttpServletResponse response) **throws** ServletException, IOException {

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

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

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

//create Object of UserDetails

UserDetails user=**new** UserDetails();

user.setId(uid);

user.setUsername(uname);

//call register method

register(user);

}

}

1. Create class od userdetails

**package** beans;

**public** **class** UserDetails {

**private** **int** id;

**private** String username;

**public** **int** getId() {

**return** id;

}

**public** **void** setId(**int** id) {

**this**.id = id;

}

**public** String getUsername() {

**return** username;

}

**public** **void** setUsername(String username) {

**this**.username = username;

}

}