**Integrating JSP, Servlet and Hibernate in an MVC application**

**Project Description**

we are going to create a real time MVC application by using Servlet, JSP and Hibernate. This example is developed using MVC pattern, where JSP pages acts as view layer, Hibernate and Oracle database as the model and servlet as controller layer. It’s a very simple and basic application.

we are going to create a web application by using servlet, jsp and hibernate. When user will access the application ,login page will be displayed. For first time user there is a link for the registration. Once user will register his/her details will be collected in servlet and finally data will be saved into database by using hibernate.

Once user got registered , user can login to the application and use the website. If user enters wrong credential error page will be displayed.

**Look and Feel :**

We have used css and images to make pages attractive. After completion of this you can understand how css and images should be used in java based web application.

For creating the web application, we used JSP for presentation logic, Servlet class for controller layer and hibernate for database access codes.

**Tools and Technologies :**

* JDK 1.7
* Hibernate 5.2.7.Final
* Eclipse
* Oracle 11g.
* Tomcat 7.0

Create database table for storing data :

SQL query is given below. Execute the following query in ORACLE database.

**CREATE TABLE USER\_DETAILS (**

**unid DECIMAL(10,2) NOT NULL ,**

**firstName VARCHAR2(45) NOT NULL ,**

**middleName VARCHAR2(45) NOT NULL ,**

**lastName VARCHAR2(45) NOT NULL ,**

**email VARCHAR2(45) NOT NULL ,**

**userid VARCHAR2(45) NOT NULL ,**

**passwd VARCHAR2(45)NOT NULL**

**);**

ALTER TABLE USER\_DETAILS ADD (

CONSTRAINT unid\_pk PRIMARY KEY (UNID));

====================================

CREATE SEQUENCE hibernate\_sequence START WITH 1; ======================================

For creating the application in Eclipse, follow the steps mentioned below.

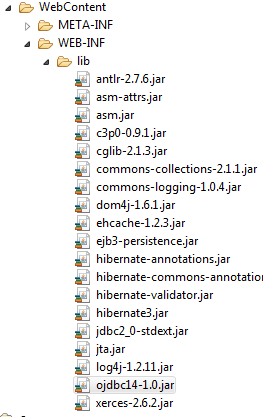
## Step 1 : Create Dynamic Web Project :

Open eclipse IDE, and go to**File -> New -> Project ->** and select **Dynamic Web Project**,specify the project name as "MVCApplication" and click on next -> finish .

Step 2 : Add Jar files for hibernate and ORACLE :

And also add

servlet-api.jar to the class path



**Directory structure of the project :**



## Step 3 : Creating web pages :

Now let us create jsp files inside Web-Content folder of your project.

**login.jsp:**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"

"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns=*"http://www.w3.org/1999/xhtml"*>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=iso-8859-1"* />

<title>Login Page</title>

<link href=*"css/style.css"* rel=*"stylesheet"* type=*"text/css"* />

</head>

<body>

<form action=*"LoginServlet"* method=*"post"*>

<div style="padding: *100px 0 0 250px*;">

<div id=*"login-box"*>

<h2>Login Page</h2>

Please provide your credential to use this website

<br>

<br>

<div id=*"login-box-name"* style="margin-top:*20px*;">User Id:</div>

<div id=*"login-box-field"* style="margin-top:*20px*;">

<input name=*"userId"* class=*"form-login"* title=*"Username"* value=*""* size=*"30"* maxlength=*"50"* />

</div>

<div id=*"login-box-name"*>Password:</div>

<div id=*"login-box-field"*>

<input name=*"password"* type=*"password"* class=*"form-login"* title=*"Password"* value=*""* size=*"30"* maxlength=*"48"* />

</div>

<br />

<span class=*"login-box-options"*>

New User? <a href=*"register.jsp"* style="margin-left:*30px*;">Register Here</a>

</span>

<br />

<br />

<input style="margin-left:*100px*;" type=*"submit"* value=*"Login"* />

</div>

</div>

</form>

</body>

</html>

**register.jsp**

<html>

<head>

<title>Registration Form</title>

<style type=*"text/css"*>

**h3**{font-family: *Calibri*; font-size: *22pt*; font-style: *normal*; font-weight: *bold*; color:*SlateBlue*;

text-align: *center*; text-decoration: *underline* }

**table**{font-family: *Calibri*; color:*white*; font-size: *11pt*; font-style: *normal*;width: *50%*;

text-align:; background-color: *SlateBlue*; border-collapse: *collapse*; border: *2px solid navy*}

**table***.inner*{border: *0px*}

</style>

</head>

<body>

<h3>Student Registration Form</h3>

<form action=*"RegisterServlet"* method=*"POST"*>

<table align=*"center"* cellpadding = *"10"*>

<tr>

<td>First Name</td>

<td><input type=*"text"* name=*"firstName"* maxlength=*"30"*/>

(max 30 characters a-z and A-Z)

</td>

</tr>

<tr>

<td>Middle Name</td>

<td><input type=*"text"* name=*"middleName"* maxlength=*"30"*/>

(max 30 characters a-z and A-Z)

</td>

</tr>

<tr>

<td>Last Name</td>

<td><input type=*"text"* name=*"lastName"* maxlength=*"30"*/>

(max 30 characters a-z and A-Z)

</td>

</tr>

<tr>

<td>Email</td>

<td><input type=*"text"* name=*"email"* maxlength=*"100"* /></td>

</tr>

<tr>

<td>User ID</td>

<td><input type=*"text"* name=*"userId"* maxlength=*"100"* /></td>

</tr>

<tr>

<td>Password</td>

<td><input type=*"text"* name=*"password"* maxlength=*"100"* /></td>

</tr>

<tr>

<td colspan=*"2"* align=*"center"*>

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

<input type=*"reset"* value=*"Reset"*>

</td>

</tr>

</table>

</form>

</body>

</html>

home.jsp:

<%@page import=*"java.util.List"*%>

<%@page import=*"com.visix.mnrao.service.LoginService"*%>

<%@page import=*"java.util.Date"*%>

<%@page import=*"com.visix.mnrao.model.User"*%>

<%@page contentType=*"text/html"* pageEncoding=*"UTF-8"*%>

<!DOCTYPE html>

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<link rel=*"stylesheet"* type=*"text/css"* href=*"css/style.css"*/>

<title>Result Page</title>

</head>

<body>

<center>

<div id=*"container"*>

<h1>Result Page</h1>

<b>This is Sample Result Page</b><br/>

<%=**new** Date()%></br>

<%

User user = (User) session.getAttribute("user");

%>

<b>Welcome <%= user.getFirstName() + " " + user.getLastName()%></b>

<br/>

<a href=*"logout.jsp"*>Logout</a>

</p>

<table>

<thead>

<tr>

<th>User ID</th>

<th>First Name</th>

<th>Middle Name</th>

<th>Last Name</th>

<th>email</th>

</tr>

</thead>

<tbody>

<%

LoginService loginService = **new** LoginService();

List<User> list = loginService.getListOfUsers();

**for** (User u : list) {

%>

<tr>

<td><%=u.getUserId()%></td>

<td><%=u.getFirstName()%></td>

<td><%=u.getMiddleName()%></td>

<td><%=u.getLastName()%></td>

<td><%=u.getEmail()%></td>

</tr>

<%}%>

<tbody>

</table>

<br/>

</div>

</center>

</body>

</html>

**logout.jsp:**

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=UTF-8"*>

<link rel=*"stylesheet"* type=*"text/css"* href=*"css/style.css"*>

<title>logout Page</title>

</head>

<body>

<%

session.removeAttribute("userId");

session.removeAttribute("password");

session.invalidate();

%>

<center>

<h1>You have successfully logged out</h1>

To login again <a href=*"login.jsp"*>click here</a>.

</center>

</body>

</html>

**error.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">

<html>

<head>

<meta http-equiv=*"Content-Type"* content=*"text/html; charset=ISO-8859-1"*>

<title>Error Page</title>

</head>

<body>

<center>

<h1>Your Login Was Unsuccessful - Please Try Again</h1>

To login again <a href=*"login.jsp"*>click here</a>.

</center>

</body>

</html>

Step 4 : Create css file

Create a folder css inside WebContent of the project. Now create a css file style.css inside this folder and add below code into this file.

**style.css:**

**body**{

font-family:*"Lucida Grande",* *"Lucida Sans Unicode",* *Verdana,* *Arial,* *Helvetica,* *sans-serif*;

font-size:*12px*;

}

**p,** **h1,** **form,** **button**{border:*0*; margin:*0*; padding:*0*;}

*.spacer*{clear:*both*; height:*1px*;}

*.mainform*{

margin:*0* *auto*;

width:*400px*;

padding:*14px*;

}

*#container*{

border:*solid* *2px* *burlywood*;

background: *silver*;

border-radius: *5px*;

}

*#container* **h1** {

font-size:*14px*;

font-weight:*bold*;

margin-bottom:*8px*;

}

*#container* **p**{

font-size:*11px*;

color:*#666666*;

margin-bottom:*20px*;

border-bottom:*solid* *1px* *burlywood*;

padding-bottom:*10px*;

}

*#container* **label**{

display:*block*;

font-weight:*bold*;

text-align:*right*;

width:*140px*;

float:*left*;

}

*#container* *.small*{

color:*#666666*;

display:*block*;

font-size:*11px*;

font-weight:*normal*;

text-align:*right*;

width:*140px*;

}

*#container* **input**{

float:*left*;

font-size:*12px*;

padding:*4px* *2px*;

border:*solid* *1px* *burlywood*;

width:*200px*;

margin:*2px* *0* *20px* *10px*;

border-radius: *5px*;

}

*#container* **button**{

clear:*both*;

margin-left:*150px*;

width:*125px*;

height:*31px*;

background:*#666666*;

text-align:*center*;

line-height:*31px*;

color:*#FFFFFF*;

font-size:*11px*;

font-weight:*bold*;

border-radius: *5px*;

}

**table,** **tr,** **td**{

border-radius: *5px*;

border:*solid* *1px*;

padding: *5px*;

width: *max-content*;

}

*#login-box* {

width:*333px*;

height: *352px*;

padding: *58px* *76px* *0* *76px*;

color: *#ebebeb*;

font: *12px* *Arial,* *Helvetica,* *sans-serif*;

margin-top: *-70px*;

background: *url(../images/login-box-backg.png)* *no-repeat* *left* *top*;

}

*#login-box* **img** {

border:*none*;

}

*#login-box* **h2** {

padding:*0*;

margin:*0*;

color: *#ebebeb*;

font: *bold* *44px* *"Calibri",* *Arial*;

}

*#login-box-name* {

float: *left*;

display:*inline*;

width:*80px*;

text-align: *right*;

padding: *14px* *10px* *0* *0*;

margin:*0* *0* *7px* *0*;

}

*#login-box-field* {

float: *left*;

display:*inline*;

width:*230px*;

margin:*0*;

margin:*0* *0* *7px* *0*;

}

*.form-login* {

width: *205px*;

padding: *10px* *4px* *6px* *3px*;

border: *1px* *solid* *#0d2c52*;

background-color:*#1e4f8a*;

font-size: *16px*;

color: *#ebebeb*;

}

*.login-box-options* {

clear:*both*;

padding-left:*87px*;

font-size: *11px*;

}

*.login-box-options* **a** {

color: *#ebebeb*;

font-size: *11px*;

}

**Step 5 : Creating hibernate persistence Class :**

Create a package **com.visix.mnrao.model** and create a java class User in this package and add following code in this class.

**User.java**

**package** com.visix.mnrao.model;

**import** java.io.Serializable;

**import** javax.persistence.Entity;

**import** javax.persistence.GeneratedValue;

**import** javax.persistence.Id;

**import** javax.persistence.Table;

@Entity

@Table(name = "USER\_DETAILS")

**public** **class** User **implements** Serializable {

@Id

@GeneratedValue

**private** Long unid;

**private** String firstName;

**private** String middleName;

**private** String lastName;

**private** String email;

**private** String userId;

**private** String passwd;

**public** User() {

}

**public** User(String firstName, String middleName, String lastName,

String email, String userId, String passwd) {

**this**.firstName = firstName;

**this**.middleName = middleName;

**this**.lastName = lastName;

**this**.email = email;

**this**.userId = userId;

**this**.passwd = passwd;

}

**public** Long getUnid() {

**return** unid;

}

**public** **void** setUnid(Long unid) {

**this**.unid = unid;

}

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getMiddleName() {

**return** middleName;

}

**public** **void** setMiddleName(String middleName) {

**this**.middleName = middleName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

**public** String getUserId() {

**return** userId;

}

**public** **void** setUserId(String userId) {

**this**.userId = userId;

}

**public** String getPasswd() {

**return** passwd;

}

**public** **void** setPasswd(String passwd) {

**this**.passwd = passwd;

}

}

**Step 5 : Create Util class for getting SeesionFactory Object**

Create a java class HibernateUtil. Method inside this class will return SessionFactory Object. SessionFactory is heavy weight object. For better performance only one SessionFactory should be created for entire application. So this class should be created as singleton.

HibernateUtil.java:

**package** com.visix.mnrao.util;

**import** org.hibernate.Session;

**import** org.hibernate.SessionFactory;

**import** org.hibernate.cfg.AnnotationConfiguration;

**public** **class** HibernateUtil {

**private** **static** **final** SessionFactory ***sessionFactory***;

**static** {

**try** {

***sessionFactory*** = **new** AnnotationConfiguration().configure("hibernate.cfg.xml").buildSessionFactory();

} **catch** (Throwable ex) {

System.***err***.println("Initial SessionFactory creation failed." + ex);

**throw** **new** ExceptionInInitializerError(ex);

}

}

**public** **static** Session openSession() {

**return** ***sessionFactory***.openSession();

}

}

**Step 6 : 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**

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

<!DOCTYPE hibernate-configuration PUBLIC "-//Hibernate/Hibernate Configuration DTD 3.0//EN" "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

<session-factory>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.Oracle9Dialect</property>

<property name=*"hibernate.connection.driver\_class"*>oracle.jdbc.driver.OracleDriver</property>

<property name=*"hibernate.connection.url"*>jdbc:oracle:thin:@localhost:1521:ORCL</property>

<property name=*"hibernate.connection.username"*>scott</property>

<property name=*"hibernate.connection.password"*>tiger</property>

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

<property name=*"hbm2ddl.auto"*>update</property>

<mapping class=*"com.visix.mnrao.model.User"* />

</session-factory>

</hibernate-configuration>

**Step 7 : Create Controller class**

Here we are going to create two classes LoginServlet and RegisterServlet which will help interaction between view layer, and business service classes.LoginSrvlet will control login request and RegisterServlet will control /new user registration process.

**LoginServlet.java:**

**package** com.visix.mnrao.ctrl;

**import** java.io.IOException;

**import** javax.servlet.ServletException;

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

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

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

**import** com.visix.mnrao.model.User;

**import** com.visix.mnrao.service.LoginService;

**public** **class** LoginServlet **extends** HttpServlet {

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

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

String userId = request.getParameter("userId");

String password = request.getParameter("password");

LoginService loginService = **new** LoginService();

**boolean** result = loginService.authenticateUser(userId, password);

User user = loginService.getUserByUserId(userId);

**if**(result == **true**){

request.getSession().setAttribute("user", user);

response.sendRedirect("home.jsp");

}

**else**{

response.sendRedirect("error.jsp");

}

}

}

**RegisterServlet.java**

**package** com.visix.mnrao.ctrl;

**import** java.io.IOException;

**import** java.io.PrintWriter;

**import** javax.servlet.ServletException;

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

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

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

**import** com.visix.mnrao.model.User;

**import** com.visix.mnrao.service.RegisterService;

**public** **class** RegisterServlet **extends** HttpServlet {

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

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

response.setContentType("text/html;charset=UTF-8");

PrintWriter out = response.getWriter();

String firstName = request.getParameter("firstName");

String middleName = request.getParameter("middleName");

String lastName = request.getParameter("lastName");

String email = request.getParameter("email");

String userId = request.getParameter("userId");

String password = request.getParameter("password");

User user = **new** User(firstName,middleName,lastName, email,userId, password);

**try** {

RegisterService registerService = **new** RegisterService();

**boolean** result = registerService.register(user);

out.println("<html>");

out.println("<head>");

out.println("<title>Registration Successful</title>");

out.println("</head>");

out.println("<body>");

out.println("<center>");

**if**(result){

out.println("<h1>Thanks for Registering with us :</h1>");

out.println("To login with new UserId and Password<a href=login.jsp>Click here</a>");

}**else**{

out.println("<h1>Registration Failed</h1>");

out.println("To try again<a href=register.jsp>Click here</a>");

}

out.println("</center>");

out.println("</body>");

out.println("</html>");

} **finally** {

out.close();

}

}

}

**Step 8 : Create Service class for Business logic**

Create two classes LoginService and RegisterService. These classes will interact with model layer for business processing.

**LoginService.java:**

**package** com.visix.mnrao.service;

**import** java.util.ArrayList;

**import** java.util.List;

**import** org.hibernate.Query;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**import** com.visix.mnrao.util.HibernateUtil;

**import** com.visix.mnrao.model.User;

**public** **class** LoginService {

**public** **boolean** authenticateUser(String userId, String password) {

User user = getUserByUserId(userId);

**if**(user!=**null** && user.getUserId().equals(userId) && user.getPasswd().equals(password)){

**return** **true**;

}**else**{

**return** **false**;

}

}

**public** User getUserByUserId(String userId) {

Session session = HibernateUtil.*openSession*();

Transaction tx = **null**;

User user = **null**;

**try** {

tx = session.getTransaction();

tx.begin();

Query query = session.createQuery("from User where userId='"+userId+"'");

user = (User)query.uniqueResult();

tx.commit();

} **catch** (Exception e) {

**if** (tx != **null**) {

tx.rollback();

}

e.printStackTrace();

} **finally** {

session.close();

}

**return** user;

}

**public** List<User> getListOfUsers(){

List<User> list = **new** ArrayList<User>();

Session session = HibernateUtil.*openSession*();

Transaction tx = **null**;

**try** {

tx = session.getTransaction();

tx.begin();

list = session.createQuery("from User").list();

tx.commit();

} **catch** (Exception e) {

**if** (tx != **null**) {

tx.rollback();

}

e.printStackTrace();

} **finally** {

session.close();

}

**return** list;

}

}

RegisterService.java:

**package** com.visix.mnrao.service;

**import** org.hibernate.Query;

**import** org.hibernate.Session;

**import** org.hibernate.Transaction;

**import** com.visix.mnrao.util.HibernateUtil;

**import** com.visix.mnrao.model.User;

**public** **class** RegisterService {

**public** **boolean** register(User user){

Session session = HibernateUtil.*openSession*();

**if**(isUserExists(user)) **return** **false**;

Transaction tx = **null**;

**try** {

tx = session.getTransaction();

tx.begin();

session.saveOrUpdate(user);

tx.commit();

} **catch** (Exception e) {

**if** (tx != **null**) {

tx.rollback();

}

e.printStackTrace();

} **finally** {

session.close();

}

**return** **true**;

}

**public** **boolean** isUserExists(User user){

Session session = HibernateUtil.*openSession*();

**boolean** result = **false**;

Transaction tx = **null**;

**try**{

tx = session.getTransaction();

tx.begin();

Query query = session.createQuery("from User where userId='"+user.getUserId()+"'");

User u = (User)query.uniqueResult();

tx.commit();

**if**(u!=**null**) result = **true**;

}**catch**(Exception ex){

**if**(tx!=**null**){

tx.rollback();

}

}**finally**{

session.close();

}

**return** result;

}

}

**Step 9: Create web.xml file :**

Create web.xml file inside WEB-INF directory of project.

**web.xml:**

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

<web-app xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"* xmlns=*"http://java.sun.com/xml/ns/javaee"* xmlns:web=*"http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"* xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd"* id=*"WebApp\_ID"* version=*"2.5"*>

<display-name>MVCApplication</display-name>

<servlet>

<display-name>LoginServlet</display-name>

<servlet-name>LoginServlet</servlet-name>

<servlet-class>com.visix.mnrao.ctrl.LoginServlet</servlet-class>

</servlet>

<servlet>

<display-name>RegisterServlet</display-name>

<servlet-name>RegisterServlet</servlet-name>

<servlet-class>com.visix.mnrao.ctrl.RegisterServlet</servlet-class>

</servlet>

<servlet-mapping>

<servlet-name>LoginServlet</servlet-name>

<url-pattern>/LoginServlet</url-pattern>

</servlet-mapping>

<servlet-mapping>

<servlet-name>RegisterServlet</servlet-name>

<url-pattern>/RegisterServlet</url-pattern>

</servlet-mapping>

<welcome-file-list>

<welcome-file>login.jsp</welcome-file>

</welcome-file-list>

</web-app>