

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app version="2.5" id="WebApp_ID"
xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd"
xmlns="http://java.sun.com/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
<display-name>MySevlet</display-name>
```

```
<welcome-file-list>
<welcome-file>index.html</welcome-file>
<welcome-file>index.htm</welcome-file>
<welcome-file>index.jsp</welcome-file>
<welcome-file>default.html</welcome-file>
<welcome-file>default.htm</welcome-file>
<welcome-file>default.jsp</welcome-file>
</welcome-file-list>
```

```
<servlet>
<description/>
<display-name>AddServlet</display-name>
<servlet-name>AddServlet</servlet-name>
<servlet-class>com.add.AddServlet</servlet-class>
</servlet>
```

```
<servlet-mapping>
<servlet-name>AddServlet</servlet-name>
<url-pattern>/MyAdd</url-pattern>
</servlet-mapping>
</web-app>
```

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    <h1 align="center">Sign In</h1>
    <div align="center">
        <form action="LogInServlet" method="post">
            User Name: <input type="text" name="username" />
            Password: <input type="text" name="password" />
            <input type="submit" value="LogIn">
        </form>
        <a href="Enroll.jsp">Enroll Here</a>
    </div>
</body>
</html>
```

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    <h1 align="center">New User Add Page</h1>
    <div align="center">
        <form action="AddServlet" method="post">
            User Name: <input type="text" name="uname" />
            Password: <input type="text" name="pword" />
            <input type="submit" value="add">
        </form>
    </div>
</body>
</html>
```

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
    <h1 align="center"> LOGIN SUSSESS! </h1>
    <div align="center">
        <a href="index.jsp">
            <input type="submit" value="LogOut">
        </a>
    </div>

</body>
</html>
```

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE hibernate-configuration SYSTEM "http://www.hibernate.org/dtd/hibernate-
configuration-3.0.dtd">
<hibernate-configuration>
<session-factory>

<!-- Database connection settings -->
<property name="connection.driver_class">com.mysql.jdbc.Driver</property>
<property name="connection.url">jdbc:mysql://localhost:3306/users</property>
<property name="connection.username">root</property>
<property name="connection.password">Mossie123?</property>
<property name="connection.pool_size">100</property>
<property name="currnt_session_context_class">thread</property>
<property
name="cache.provider_class">org.hibernate.cache.internal.NoCacheProvider</property>
<property name="show_sql">true</property>
<property name="hbm2ddl.auto">update</property>
<mapping class="com.to.Client"/>
</session-factory>
</hibernate-configuration>

```

```

package com.to;

```

```

import javax.persistence.*;

@Entity
@Table(name = "user")
public class Client {
    @Id
    @Column(name= "username", unique = true)
    private String userName;
    @Column(name = "password")
    private String password;

    public Client() {
        super();
    }
    public Client(String userName, String password) {
        super();
        this.userName = userName;
        this.password = password;
    }
    public String getUserName() {
        return userName;
    }
    public void setUserName(String userName) {
        this.userName = userName;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}

```

```

package com.log;

import java.io.IOException;
import java.io.PrintWriter;

import javax.persistence.NoResultException;
import javax.persistence.criteria.CriteriaBuilder;
import javax.persistence.criteria.CriteriaQuery;
import javax.persistence.criteria.Root;
import javax.servlet.RequestDispatcher;
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.Criteria.*;
import org.hibernate.*;
import org.hibernate.query.Query;
import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.Transaction;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;

import com.to.Client;

/**
 * Servlet implementation class LogInServlet
 */
@WebServlet("/LogInServlet")
public class LogInServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public LogInServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        try {
            //these lines of code can be in a Model Class to use to open a
            session for multiple calls,
            //dining philosophers problem would have to be solved to use it
            properly.

            Configuration config = new Configuration().configure();
            config.addAnnotatedClass(com.to.Client.class);

```

```

        StandardServiceRegistryBuilder builder = new
StandardServiceRegistryBuilder().applySettings(config.getProperties());
        SessionFactory factory =
config.buildSessionFactory(builder.build());
        Session session = factory.openSession();
        //-----
        String password = request.getParameter("password");
        String username = request.getParameter("username");
        //these lines can also be made in the Model to allow for
different query calls to respective methods
        CriteriaBuilder cr = session.getCriteriaBuilder();
        CriteriaQuery<Client> query = cr.createQuery(Client.class);
        Root<Client> root = query.from(Client.class);
        query.select(root).where(cr.equal(root.get("userName"),
username));
        Query<Client> q = session.createQuery(query);
        Client client = q.getSingleResult();
        //-----
        if(client.getPassword().equals(password)) {
            RequestDispatcher rd =
request.getRequestDispatcher("Acct.jsp");
            rd.forward(request, response);
        }

        session.close();

    }catch(NoResultException e) {
        PrintWriter out = response.getWriter();
        RequestDispatcher rd= request.getRequestDispatcher("index.jsp");
        rd.include(request, response);
        out.print("<center><span style='color:red'>Please try again:
INVALID ENTRY</span></center>");
        e.printStackTrace();
    }catch(Exception e) {
        e.printStackTrace();
    }
}
}

```



```

package com.add;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.RequestDispatcher;
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.Transaction;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;

import com.to.Client;

/**
 * Servlet implementation class AddServlet
 */
@WebServlet("/AddServlet")
public class AddServlet extends HttpServlet {
    private static final long serialVersionUID = 1L;

    /**
     * @see HttpServlet#HttpServlet()
     */
    public AddServlet() {
        super();
        // TODO Auto-generated constructor stub
    }

    /**
     * @see HttpServlet#service(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void service(HttpServletRequest request, HttpServletResponse
    response) throws ServletException, IOException {
        try {
            Configuration config = new Configuration().configure();
            config.addAnnotatedClass(com.to.Client.class);
            StandardServiceRegistryBuilder builder = new
            StandardServiceRegistryBuilder().applySettings(config.getProperties());
            SessionFactory factory =
            config.buildSessionFactory(builder.build());

            PrintWriter out = response.getWriter();
            String password = request.getParameter("pword");
            String username = request.getParameter("uname");

            if((username.equalsIgnoreCase("") || username == null) ||
            (password.equalsIgnoreCase("") || password == null)) {

```

```

        out.println("<h1 align='center'><span
style='color:red'>Item Not Added</span></h1>");
    }else {

        Session session = factory.openSession();
        Transaction trans = session.beginTransaction();
        Client p1 = new Client(username, password);
        session.save(p1);
        trans.commit();
        session.close();

        RequestDispatcher rd = request.getRequestDispatcher("index.jsp");
        rd.forward(request, response);
    } //end of 'if statement'
} catch (NumberFormatException e) {
    e.printStackTrace();
} catch (Exception e) {
    e.printStackTrace();
}

}

}

```