CSCI 399

Assignment 6

Student name: LiZhao Xiaoxiong

Student number:4520841

Table of Contents
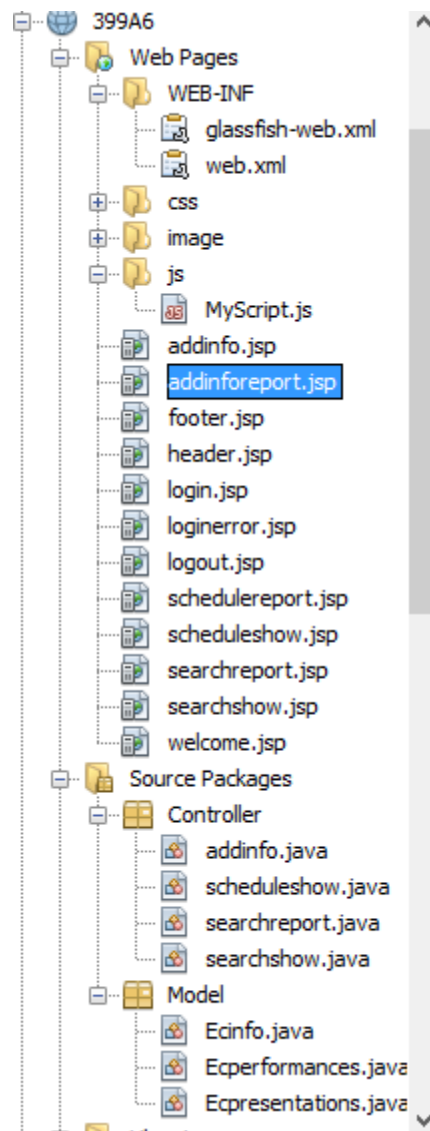
Part 1 Introduction

Well this assignment is rebuild the application from assignment 5, but use more wise way to implement, like use JPA to substitute JDBC, and use JSP page rather than HTML page, the good thing about JSP page is we can use JSTL. All things is tag looks more structure and beautiful.

Part 2 Overview

Project directory structure

```
└── 399A6
    └── Web Pages
        └── WEB-INF
            ├── glassfish-web.xml
            ├── web.xml
        ├── css
        ├── image
        └── js
            └── MyScript.js
        ├── addinfo.jsp
        ├── addinforeport.jsp
        ├── footer.jsp
        ├── header.jsp
        ├── login.jsp
        ├── loginerror.jsp
        ├── logout.jsp
        ├── schedulereport.jsp
        ├── scheduleshow.jsp
        ├── searchreport.jsp
        ├── searchshow.jsp
        └── welcome.jsp
    └── Source Packages
        └── Controller
            ├── addinfo.java
            ├── scheduleshow.java
            ├── searchreport.java
            └── searchshow.java
        └── Model
            ├── Ecinfo.java
            ├── Ecperformances.java
            └── Ecpresentations.java
```

And the datasource still use the same one as Ass5

Edit JDBC Resource

Edit an existing JDBC data source.

[Load Defaults]

JNDI Name:	jdbc/entertaimentDatasource
Pool Name:	EconnectionPool ▼
	Use the JDBC Connection Pools page to create new pools
Deployment Order:	100
	Specifies the loading order of the resource at server startup. Lower numbers are loaded first.
Description:	local mysql
Status:	☑ Enabled

Edit JDBC Connection Pool Properties

Modify properties of an existing JDBC connection pool.

Pool Name: EconnectionPool

Additional Properties (3)

[Add Property] [Delete Properties]

Select	Name	Value	Description
☐	URL	jdbc:mysql://localhost:3306/entertainment_cen	
☐	User	root	
☐	Password	sasakixi	

glassfish-resources.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE resources PUBLIC "-//GlassFish.org//DTD GlassFish Application Server 3.1 Resource
Definitions//EN" "http://glassfish.org/dtds/glassfish-resources_1_5.dtd">
<resources>
  <jdbc-resource enabled="true" jndi-name="jdbc/entertaimentDatasource" object-type="user"
pool-name="EconnectionPool">
    <description>local mysql</description>
  </jdbc-resource>

  <jdbc-connection-pool allow-non-component-callers="false" associate-with-thread="false"
connection-creation-retry-attempts="0" connection-creation-retry-interval-in-seconds="10"
connection-leak-reclaim="false" connection-leak-timeout-in-seconds="0" connection-validation-
method="auto-commit" datasource-
classname="com.mysql.jdbc.jdbc2.optional.MysqlDataSource" fail-all-connections="false" idle-
timeout-in-seconds="300" is-connection-validation-required="false" is-isolation-level-
guaranteed="true" lazy-connection-association="false" lazy-connection-enlistment="false"
match-connections="false" max-connection-usage-count="0" max-pool-size="32" max-wait-time-
in-millis="60000" name="EconnectionPool" non-transactional-connections="false" pool-resize-
quantity="2" res-type="javax.sql.DataSource" statement-timeout-in-seconds="-1" steady-pool-
size="8" validate-atmost-once-period-in-seconds="0" wrap-jdbc-objects="false">
    <property name="URL" value="jdbc:mysql://localhost:3306/entertainment_centre"/>
```

```xml
        <property name="User" value="root"/>
        <property name="Password" value="sasakixi"/>
    </jdbc-connection-pool>
</resources>
```

And access control is still the same as Ass5:

Properties specific to this Class

JAAS Context: *	jdbcRealm
	Identifier for the login module to use for this realm
JNDI: *	jdbc/entertaimentDatasource
	JNDI name of the JDBC resource used by this realm
User Table: *	entertainment_centre.ecuserroles
	Name of the database table that contains the list of authorized users for this realm
User Name Column: *	username
	Name of the column in the user table that contains the list of user names
Password Column: *	pwd
	Name of the column in the user table that contains the user passwords
Group Table: *	entertainment_centre.ecuserroles
	Name of the database table that contains the list of groups for this realm
Group Table User Name Column:	username
	Name of the column in the user group table that contains the list of groups for this realm
Group Name Column: *	role
	Name of the column in the group table that contains the list of group names
Password Encryption Algorithm: *	MD5
	This denotes the algorithm for encrypting the passwords in the database. It is a security risk t
Assign Groups:	
	Comma-separated list of group names
Database User:	
	Specify the database user name in the realm instead of the JDBC connection pool
Database Password:	
	Specify the database password in the realm instead of the JDBC connection pool
Digest Algorithm:	MD5
	Digest algorithm (default is SHA-256); note that the default was MD5 in GlassFish versions p
Encoding:	
	Encoding (allowed values are Hex and Base64)
Charset:	

And this one role group mapping

glassfish-web.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE glassfish-web-app PUBLIC "-//GlassFish.org//DTD GlassFish Application Server 3.1
Servlet 3.0//EN" "http://glassfish.org/dtds/glassfish-web-app_3_0-1.dtd">
<glassfish-web-app error-url="">
    <context-root>399A6</context-root>
```

```xml
        <security-role-mapping>
            <role-name>detailers</role-name>
            <group-name>clerical</group-name>
        </security-role-mapping>
        <security-role-mapping>
            <role-name>schedulers</role-name>
            <group-name>events</group-name>
        </security-role-mapping>
        <class-loader delegate="true"/>
        <jsp-config>
            <property name="keepgenerated" value="true">
                <description>Keep a copy of the generated servlet class' java code.</description>
            </property>
        </jsp-config>
</glassfish-web-app>
```

And one more file which contain all kinds of deployment things

web.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<web-app               version="3.1"                xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_3_1.xsd">
    <servlet>
        <servlet-name>searchshow</servlet-name>
        <servlet-class>Controller.searchshow</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>addinfo</servlet-name>
        <servlet-class>Controller.addinfo</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>scheduleshow</servlet-name>
        <servlet-class>Controller.scheduleshow</servlet-class>
    </servlet>
    <servlet>
        <servlet-name>searchreport</servlet-name>
        <servlet-class>Controller.searchreport</servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>searchshow</servlet-name>
```

```xml
            <url-pattern>/searchshow</url-pattern>
        </servlet-mapping>
        <servlet-mapping>
            <servlet-name>addinfo</servlet-name>
            <url-pattern>/addinfo</url-pattern>
        </servlet-mapping>
        <servlet-mapping>
            <servlet-name>scheduleshow</servlet-name>
            <url-pattern>/scheduleshow</url-pattern>
        </servlet-mapping>
        <servlet-mapping>
            <servlet-name>searchreport</servlet-name>
            <url-pattern>/searchreport</url-pattern>
        </servlet-mapping>
        <session-config>
            <session-timeout>
                30
            </session-timeout>
        </session-config>
        <welcome-file-list>
            <welcome-file>welcome.jsp</welcome-file>
        </welcome-file-list>
        <security-constraint>
            <display-name>detailing constraint</display-name>
            <web-resource-collection>
                <web-resource-name>event info constraint</web-resource-name>
                <description>who can add supp info</description>
                <url-pattern>/addinfo</url-pattern>
            </web-resource-collection>
            <auth-constraint>
                <description>what kind of people can add supplement info</description>
                <role-name>detailers</role-name>
                <role-name>schedulers</role-name>
            </auth-constraint>
        </security-constraint>
        <security-constraint>
            <display-name>schedule constraint</display-name>
            <web-resource-collection>
                <web-resource-name>events schedule constrains</web-resource-name>
                <description>who can add event fo database</description>
                <url-pattern>/scheduleshow</url-pattern>
            </web-resource-collection>
            <auth-constraint>
                <description>who can add events to db</description>
```

```xml
                <role-name>schedulers</role-name>
            </auth-constraint>
        </security-constraint>
        <login-config>
            <auth-method>FORM</auth-method>
            <realm-name>entertaiment_certificate</realm-name>
            <form-login-config>
                <form-login-page>/login.jsp</form-login-page>
                <form-error-page>/loginerror.jsp</form-error-page>
            </form-login-config>
        </login-config>
        <security-role>
            <description>add shows and add supplement data</description>
            <role-name>schedulers</role-name>
        </security-role>
        <security-role>
            <description>only can add supplement data</description>
            <role-name>detailers</role-name>
        </security-role>
</web-app>
```

The main difference is there one more xml file

persistence.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<persistence        version="2.1"        xmlns="http://xmlns.jcp.org/xml/ns/persistence"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/persistence
http://xmlns.jcp.org/xml/ns/persistence/persistence_2_1.xsd">
   <persistence-unit name="399A6PU" transaction-type="JTA">
      <jta-data-source>jdbc/entertaimentDatasource</jta-data-source>
      <exclude-unlisted-classes>false</exclude-unlisted-classes>
      <properties/>
   </persistence-unit>
</persistence>
```
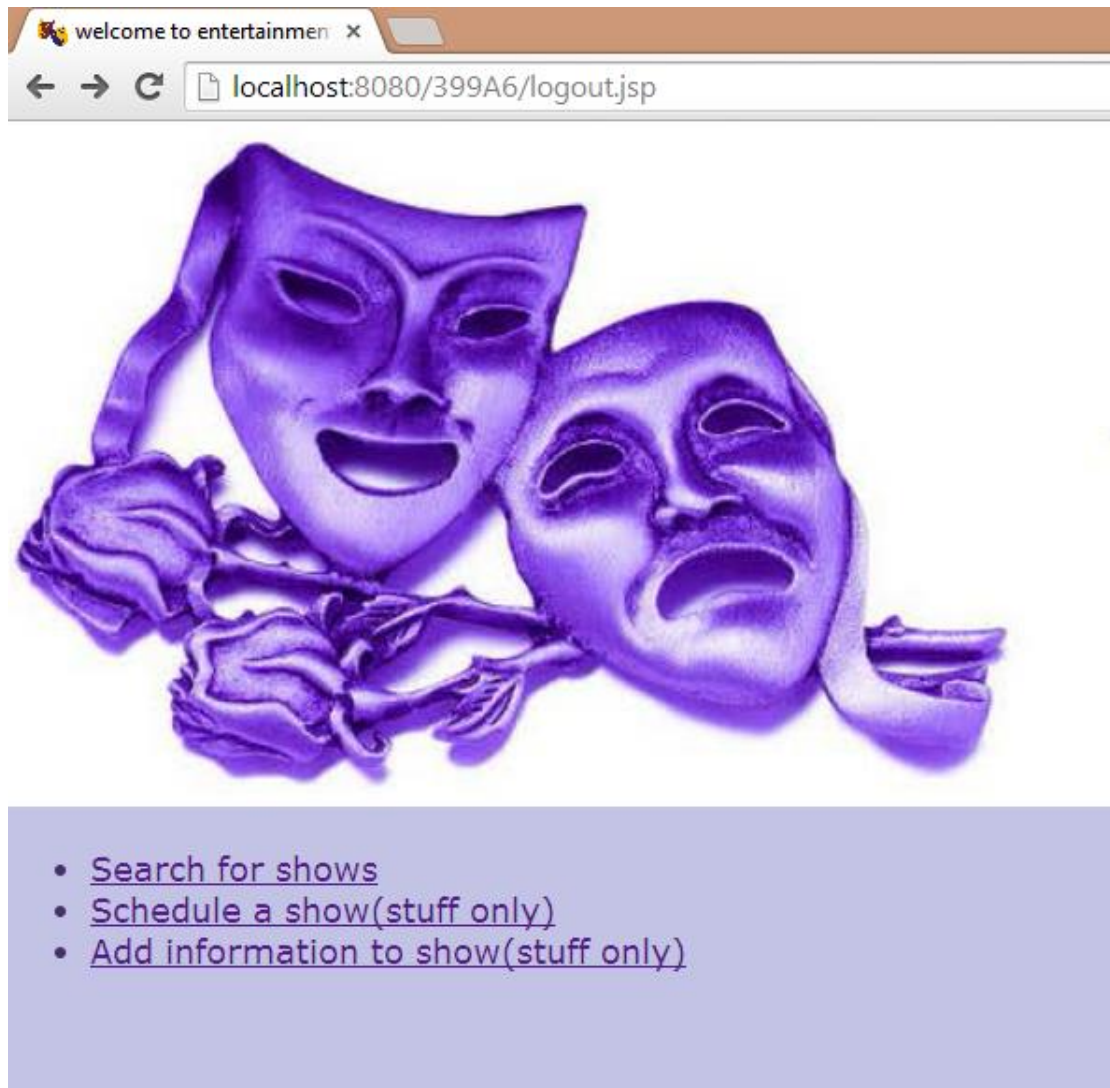
Part 3 Demonstration of working site

Let still start with searching first, there are a lot of data waiting for searching because of Assignment 4 and Assignment 5.Let search Film genre, this one is new add in Assignment 5:
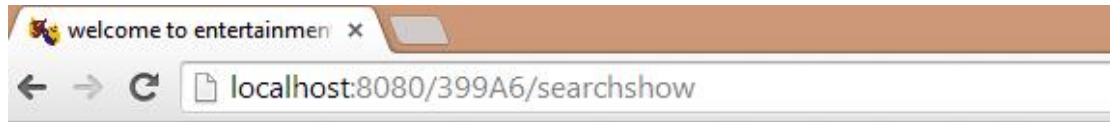

Same old Same old welcome page

Search for the show Film in studio and in whole May day

Yeah this is what I'm talking about

Shows on at entertainment center

- [Avengers: Age of Ultron](#)

Nice and sweet, the details for the show

Ok now testing wide if time can affect getting shows: start with sort of small range of time first:

Ok just one show for this type during April and May

Now let's do wide range of time from March to May
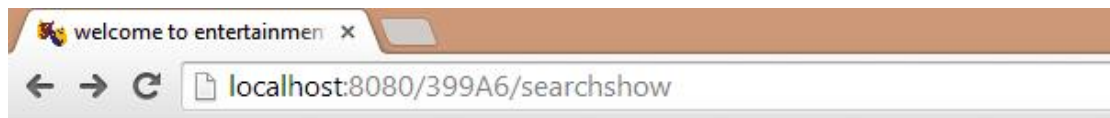
Finally got more Opera
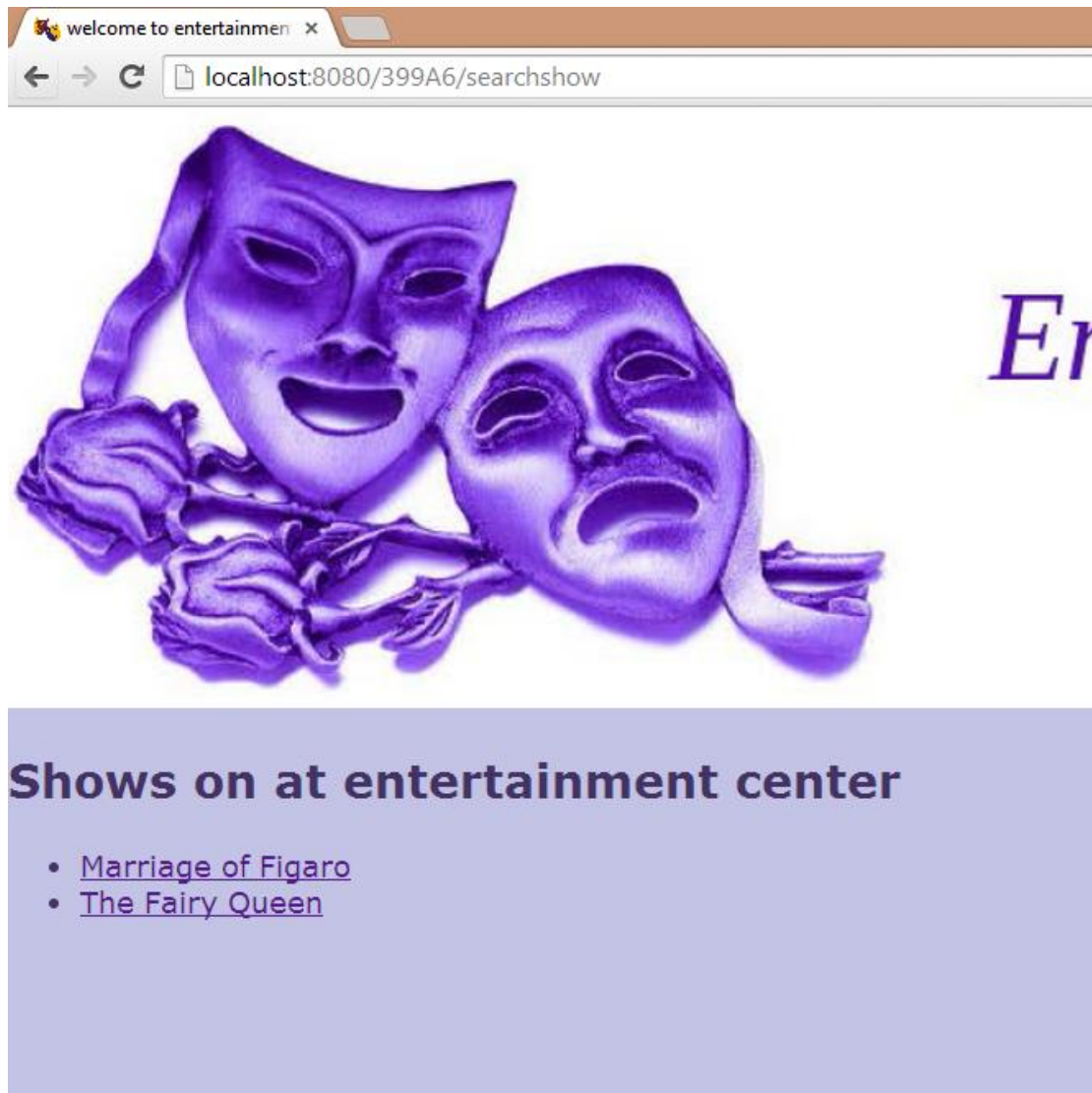
Ok Let's add more data, How about one more Film and also demonstrate role access control

Same as Ass5, scheduler do events and supplement data and detailers can only add supplement data

			username	fullname	pwd	role
☐ ✎ Edit ⯃ Copy ⊖ Delete			littlemad	mad davenport	5e8fdcc9d884a9a5eba142c0a03402b8	events
☐ ✎ Edit ⯃ Copy ⊖ Delete			victor	Xiaoxiong LiZhao	f647e02a69ab0e51780373f86f89a12a	clerical

Let's login as scheduler first, click constraint page, intercepted by login page:

welcome to entertainmen ×

localhost:8080/399A6/scheduleshow
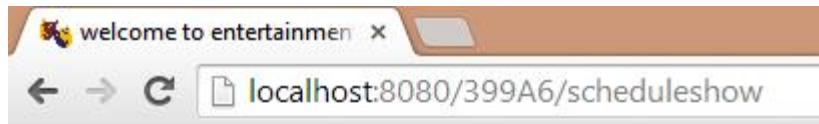
Enter your name and password

Name:	littlemad
Password:	••••
	OK

Yeah the schedule form represent to us:

Currently logged in as littlemad Logout

Create a new show record

┌─ Add a show ───
│ Show identifier
│ [   ]
│ Title
│ [   ]
│ Venue
│ [Opera   ▼]
│ Show type
│ [Drama   ▼]
│ From date
│ [mm/dd/yyyy   ]
│ To date
│ [mm/dd/yyyy   ]
│ Company [   ]
│ Description
│ [   ]
│ [   ]
│
│ Performances
│ [   ]
│ [   ]

Let's fill this form

Film_2

Title

Terminator Genisys

Venue

Studio ▾

Show type

Film ▾

From date

05/11/2015

To date

05/29/2015

Company Skydance Productions

Description

Faced with unlikely allies and dangerous
new enemies, Reese sets out on an
unexpected new mission: reset the
future.

Performances

2015-05-12,evening;
2015-05-20,matinee;
2015-05-25,evening;

Add event

Currently logged in as littlemad **Logout**

New record added

- Schedule another event
- Add information to show
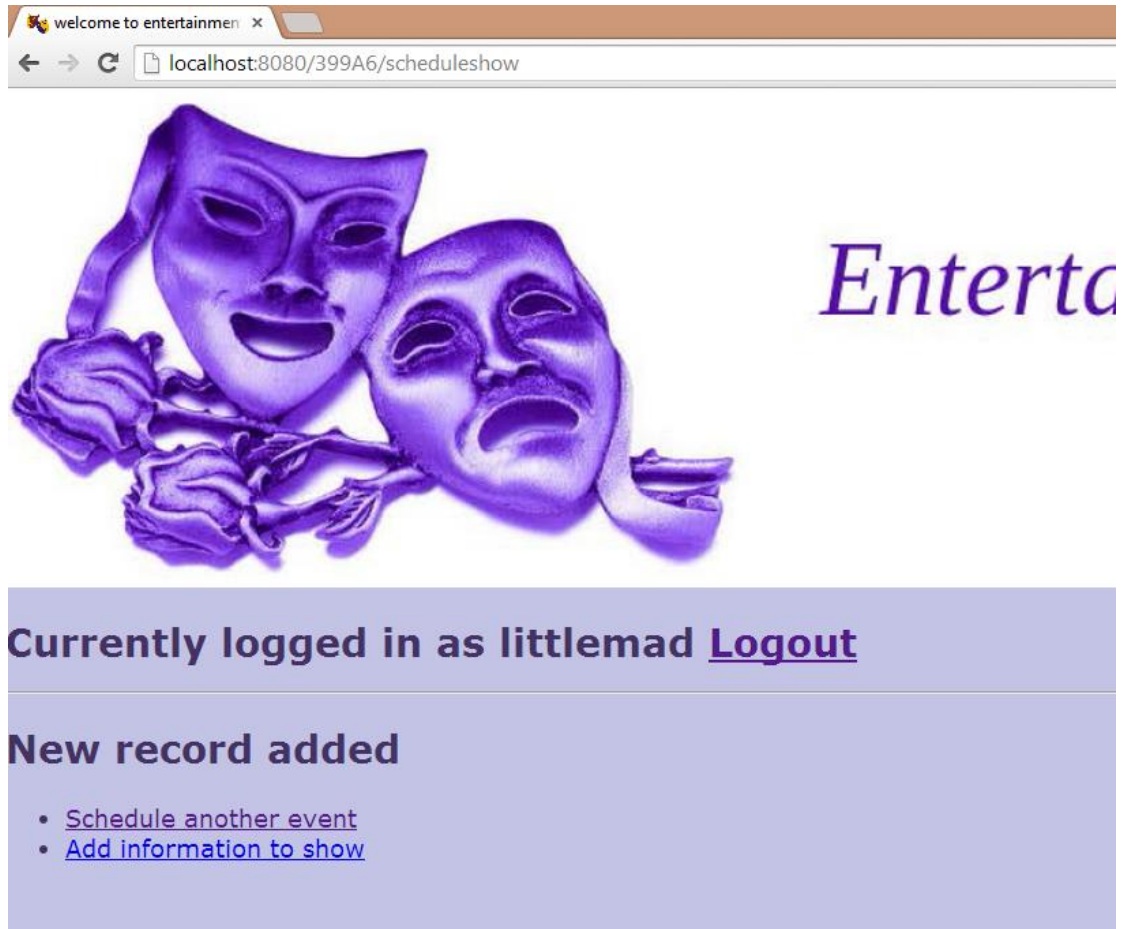
Yes new record added, to demonstrate the what role can do I will use detailer to add the supplement data, although scheduler can also add these data, but it's better to use another role to demonstrate role access control, So first logout, back to welcome page

Now login as clerical the detailers and click schedule a show:

Currently logged in as victor Logout

Sorry u are not suppose to be here

you don't have any access right in here, check the username and password

Access denied, and let's go back and click another:

welcome to entertainmen ×

localhost:8080/399A6/addinfo

Currently logged in as victor Logout

Show
Show identifier

Supplementary data
Picture Comment
Add data

Action
Add details

Yes sure he can do this

And yeah add img and comments

These data are persist in the database:

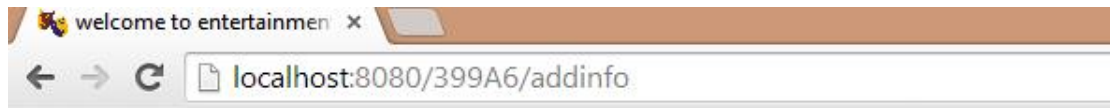|						
☐	🖉 Edit	⧉ Copy	⊖ Delete	4	comedy_1	evening	2015-04-14
☐	🖉 Edit	⧉ Copy	⊖ Delete	5	comedy_1	evening	2015-04-16
☐	🖉 Edit	⧉ Copy	⊖ Delete	6	opera_1	matinee	2015-04-11
☐	🖉 Edit	⧉ Copy	⊖ Delete	7	opera_1	evening	2015-04-14
☐	🖉 Edit	⧉ Copy	⊖ Delete	8	opera_1	evening	2015-04-16
☐	🖉 Edit	⧉ Copy	⊖ Delete	12	opera_2	matinee	2015-03-05
☐	🖉 Edit	⧉ Copy	⊖ Delete	13	opera_2	evening	2015-03-19
☐	🖉 Edit	⧉ Copy	⊖ Delete	14	opera_2	evening	2015-04-07
☐	🖉 Edit	⧉ Copy	⊖ Delete	15	comedy_2	evening	2015-04-22
☐	🖉 Edit	⧉ Copy	⊖ Delete	16	comedy_2	evening	2015-04-09
☐	🖉 Edit	⧉ Copy	⊖ Delete	17	comedy_2	matinee	2015-04-06
☐	🖉 Edit	⧉ Copy	⊖ Delete	18	jazz_1	matinee	2015-05-11
☐	🖉 Edit	⧉ Copy	⊖ Delete	19	jazz_1	matinee	2015-05-15
☐	🖉 Edit	⧉ Copy	⊖ Delete	20	jazz_1	evening	2015-05-19
☐	🖉 Edit	⧉ Copy	⊖ Delete	21	Film_1	evening	2015-05-06
☐	🖉 Edit	⧉ Copy	⊖ Delete	22	Film_1	evening	2015-05-11
☐	🖉 Edit	⧉ Copy	⊖ Delete	23	Film_2	matinee	2015-05-20
☐	🖉 Edit	⧉ Copy	⊖ Delete	24	Film_2	evening	2015-05-25
☐	🖉 Edit	⧉ Copy	⊖ Delete	25	Film_2	evening	2015-05-12

↑ ☐ Check All   With selected: 🖉 Change   ⊖ Delete   📊 E

☐	🖉 Edit	⧉ Copy	⊖ Delete	13	jazz_1	On the way to a jazz competition, Andrew's bus bre...	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	14	jazz_1	On stage at the jazz festival, Fletcher reveals th...	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	15	Film_1	great combo great team	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	16	Film_1	how many lines of code can build this machine, 100...	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	17	Film_1	nice chaos fight	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	18	Film_2	yeah T800 and sarah conner	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...
☐	🖉 Edit	⧉ Copy	⊖ Delete	19	Film_2	I wonder if programming T800 robots maybe need bil...	data:image/jpeg;base64,/9j/4AAQSkZJRgABAQAAAQABAAD...

↑ ☐ Check All   With selected: 🖉 Change   ⊖ Delete   📊 Export

Part 4 Display and explanation of code

I actually finished all the assignments one month ago, lol, I like 399, it is useful and interesting

addinfo.jsp

```
<%--
    Document    : addinfo
    Created on : 19-Apr-2015, 15:33:28
    Author      : Victor
--%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>

<c:if test="${not empty param.si && not empty param.title}">
    <c:if test="${param.si != ''&&param.title != ''}">
        <h1>
            Details added to <c:out value="${param.si}"/> : <c:out value="${param.title}"/>
        </h1>
    </c:if>
</c:if>

    <form method='post' action='addinfo'>
        <fieldset>
            <legend>Show</legend>
            <label for='si'>Show identifier</label>
            <input type='text' id='si' name='si'>
        </fieldset>

        <fieldset>
            <legend>Supplementary data</legend>
            <table border='3' id='Myshow'>
                <tr><th>Picture</th><th>Comment</th></tr>
            </table>
                <input type='button' onclick='doAdd()' value='Add data'>
        </fieldset>
```

```
        <fieldset>
            <legend>Action</legend>
                <input type='submit' value='Add details'>
        </fieldset>
    </form>
<jsp:include page="footer.jsp"/>
```

addinforeport.jsp

```
<%--
    Document    : addinforeport
    Created on : 23-Apr-2015, 00:41:23
    Author      : Victor
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>

<h1>New information added</h1>
<hr>
<ul>
    <li><a href='scheduleshow'>Schedule a    event</a></li>
    <li><a href='addinfo'>Add more information to show</a></li>
</ul>

<jsp:include page="footer.jsp"/>
```

footer.jsp

```
<%--
    Document    : footer
    Created on : 19-Apr-2015, 12:51:56
    Author      : Victor
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
</section>
    </body>
</html>
```

header.jsp

```
<%--
    Document    : header
    Created on : 19-Apr-2015, 12:12:37
    Author      : Victor
--%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
    <head>
        <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
        <title>welcome to entertainment center</title>
        <script src = 'js/MyScript.js'></script>
        <link rel="stylesheet" type="text/css" href="css/Mystyle.css" />
        <link href="image/drafavicon.ico" mce_href="favicon.ico" rel="icon" type="image/x-icon" />
    </head>
    <body>
        <header>
            <style>
                .targets{
                    width:400px;
                    height:190px;
                    background-color:#aaaaaa;
                }
            </style>
            <img src="image/layoutimg.PNG">
            <c:if test="${pageContext.request.remoteUser != null}">
                <h1>Currently logged in as <c:out value="${pageContext.request.remoteUser}"/> <a href='logout.jsp'>Logout</a></h1>
            </c:if>
        </header>
            <section>
```

login.jsp

```
<%--
    Document    : login
    Created on : 19-Apr-2015, 12:53:41
    Author      : Victor
```

--%>

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
<form method="post" action="j_security_check" >
            <label for="logintable"><b>Enter your name and password</b></label>
            <table id="logintable" border = "3" >
                <tr>
                    <th>Name:</th>
                    <td><input type="text" name="j_username"></td>
                </tr>
                <tr>
                    <th>Password:</th>
                    <td><input type="password" name="j_password"></td>
                </tr>
                <tr>
                    <td align="center" colspan="2">
                        <input type="submit"    value="OK">
                    </td>
                </tr>
            </table>
        </form>
<jsp:include page="footer.jsp"/>
```

loginerror.jsp

```
<%--
    Document     : loginerror
    Created on : 19-Apr-2015, 12:55:35
    Author       : Victor
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
<h1>Sorry u are not suppose to be here</h1>
<p><b>you don't have any access right in here, check the username and password</b></p>
<jsp:include page="footer.jsp"/>
```

logout.jsp

```jsp
<%--
    Document    : logout
    Created on : 19-Apr-2015, 13:30:20
    Author      : Victor
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<% request.logout();%>
<jsp:forward page="welcome.jsp"/>
```

schedulereport.jsp

```jsp
<%--
    Document    : schedulereport
    Created on : 22-Apr-2015, 23:57:11
    Author      : Victor
--%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
<hr>
<h1>New record added</h1>
<ul>
    <li><a href='scheduleshow'>Schedule another event</a></li>
    <li><a           href='addinfo?si=<c:out           value="${requestScope.sid}"/>&title=<c:out
value="${requestScope.title}"/>'>Add information to show</a></li>
</ul>
<jsp:include page="footer.jsp"/>
```

scheduleshow.jsp

```jsp
<%--
    Document    : scheduleshow
    Created on : 19-Apr-2015, 16:38:52
    Author      : Victor
--%>
```

```
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
<form method='post' action='scheduleshow'>
    <h1>Create a new show record</h1>
        <fieldset>
            <legend>Add a show</legend>
                <label for='showid'>Show identifier</label>
                <br><input type='text' id='showid' name='showid'><br>
                <label for='title'>Title</label>
                <br><input type='text' id='title' name='title'><br>
                <label for='venue'>Venue</label>
                <br>
                <select id='ve' name='ve'>
                    <option value='Opera'>Opera</option>
                    <option value='Concert'>Concert</option>
                    <option value='Playhouse'>Playhouse</option>
                    <option value='Studio'>Studio</option>
                </select><br>
                <label for='st'>Show type</label>
                    <br>
                <select id='st' name = 'st'>
                    <option value='Drama'>Drama</option>
                    <option value='Film'>Film</option>
                    <option value='Opera'>Opera</option>
                    <option value='Jazz'>Jazz</option>
                    <option value='World Music'>World Music</option>
                    <option value='Ballet'>Ballet</option>
                    <option value='Recital'>Recital</option>
                    <option value='Concert'>Concert</option>
                    <option value='Choral'>Choral</option>
                    <option value='Contemporary Dance'>Contemporary Dance</option>
                    <option value='Comedy'>Comedy</option>
                    <option value='Children'>Children</option>
                </select><br>
                <label for='sd'>From date</label>
                    <br><input type='date' id='sd' name = 'sd'><br>
                    <label for='ed'>To date</label>
                    <br><input type='date' id='ed' name = 'ed'><br>
                <label for='com'>Company</label>
                <input type='text' id='com' name='com'><br>
                <label for='des'>Description</label>
                <br><textarea id='des' name='des' rows='4' cols='40'></textarea><br>
```

```
                    <label for='per'>Performances</label>
                    <br><textarea id='Per' name='per' rows='4' cols='40' readonly =
'readonly'></textarea><br>
                    <input type='submit' value='Add event'><br>
            </fieldset><br>
</form><br>

<fieldset>
      <legend>Enter performance details</legend>
            <label for='st'>Show time</label>
            <br>
            <select id='stime' name='st'>
                  <option value='matinee'>matinee</option>
                  <option value='evening'>evening</option>
            </select><br>
            <label for=date>Date</label>
            <br><input type='date' id='d' name='date'><br>
            <button onclick='doaddp()'>Add performance</button><br>
</fieldset>
<jsp:include page="footer.jsp"/>
```

searchreport.jsp

```
<%--
      Document    : searchreport
      Created on : 19-Apr-2015, 22:07:51
      Author       : Victor
--%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
<h1><c:out value="${requestScope.basic.title}"/></h1>
<h2><c:out value="${requestScope.basic.mykey}"/></h2>
<h2>On    at    <c:out    value="${requestScope.basic.company}"/>    from    <c:out
value="${requestScope.basic.startseason}"/>                        to                        <c:out
value="${requestScope.basic.endseason}"/></h2>
<p><c:out value="${requestScope.basic.shortdescription}"/></p>
<h2>Scheduled performances</h2>

<table border="3">
```

```jsp
            <c:forEach var="times" items="${requestScope.time}">
                <tr>
                    <td><c:out value="${times.showtime}"/></td>
                    <fmt:formatDate value="${times.showdate}" var="formattedDate" pattern="yyyy-
MM-dd"/>
                    <td><c:out value="${formattedDate}"/></td>
                </tr>
            </c:forEach>
</table>

<table border="3">
        <c:forEach var="info" items="${requestScope.sup}">
                <tr>
                    <td><img src="<c:out value="${info.picy}"/>"></td>
                    <td><c:out value="${info.details}"/></td>
                </tr>
        </c:forEach>
</table>
<br>
<br>
<jsp:include page="footer.jsp"/>
```

searchshow.jsp

```jsp
<%--
        Document      : searchshow
        Created on : 19-Apr-2015, 13:53:46
        Author        : Victor
--%>

<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>

<c:choose>
        <c:when test="${requestScope.listshow == null}">
                <h1>what is at entertainment center</h1>
                <form method='post' action='searchshow'>
        <fieldset>
                <legend>Search for shows</legend>
                <label for='st'>Show type</label>
                <select id='st' name = 'st'>
```

```html
            <option value='Drama'>Drama</option>
            <option value='Film'>Film</option>
            <option value='Opera'>Opera</option>
            <option value='Jazz'>Jazz</option>
            <option value='World Music'>World Music</option>
            <option value='Ballet'>Ballet</option>
            <option value='Recital'>Recital</option>
            <option value='Concert'>Concert</option>
            <option value='Choral'>Choral</option>
            <option value='Contemporary Dance'>Contemporary Dance</option>
            <option value='Comedy'>Comedy</option>
            <option value='Children'>Children</option>
        </select><br>

        <label for='ve'>Venue</label>
        <select id='ve' name='ve'>
            <option value='Opera'>Opera</option>
            <option value='Concert'>Concert</option>
            <option value='Playhouse'>Playhouse</option>
            <option value='Studio'>Studio</option>
        </select><br>

        <label for='sd'>Start date</label>
        <input type='date' id='sd' name = 'sd'><br>
        <label for='ed'>End date</label>
        <input type='date' id='ed' name = 'ed'><br>
        <input type='submit' value='Search'>
    </fieldset>
    </form>
    </c:when>
    <c:when test="${requestScope.listshow != null}">
        <h1>Shows on at entertainment center</h1>
        <c:if test="${requestScope.len == 0}">
            <h2>nothing find!</h2>
        </c:if>
        <c:if test="${requestScope.len != 0}">
            <ul>
                <c:forEach var="shows" items="${requestScope.listshow}">
                    <li>
                        <a                               href="searchreport?title=<c:out
value="${shows.title}"/>"><c:out value="${shows.title}"/></a>
                    </li>
                </c:forEach>
            </ul>
```

```
        </c:if>

    </c:when>
</c:choose>


<jsp:include page="footer.jsp"/>
```

```
<%--
    Document      : welcome
    Created on : 19-Apr-2015, 11:58:13
    Author        : Victor
--%>

<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<jsp:include page="header.jsp"/>
    <ul>
        <li><a href="searchshow">Search for shows</a></li>
        <li><a href="scheduleshow">Schedule a show(stuff only)</a></li>
        <li><a href="addinfo">Add information to show(stuff only)</a></li>
    </ul>
<jsp:include page="footer.jsp"/>
```

Controller/ addinfo.java

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */

package Controller;

import Model.Ecinfo;
import Model.Ecpresentations;
import java.io.IOException;
import java.io.PrintWriter;
```

```java
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.annotation.Resource;
import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.Persistence;
import javax.persistence.PersistenceUnit;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.transaction.HeuristicMixedException;
import javax.transaction.HeuristicRollbackException;
import javax.transaction.NotSupportedException;
import javax.transaction.RollbackException;
import javax.transaction.SystemException;
import javax.transaction.UserTransaction;

/**
 *
 * @author Victor
 */
public class addinfo extends HttpServlet {

    @PersistenceUnit(unitName="399A6PU")
    private EntityManagerFactory emf;
    @Resource
    private UserTransaction utx;
    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the
left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        RequestDispatcher view = request.getRequestDispatcher("addinfo.jsp");
        view.forward(request, response);
```

```java
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
                throws ServletException, IOException {
        try {
            String showid = request.getParameter("si");

            String[] pics;
            pics = request.getParameterValues("pics[]");

            String[] comms;
            comms = request.getParameterValues("comms[]");


            utx.begin();
            EntityManager em = emf.createEntityManager();
            for(int i = 0;i < pics.length;i++){
                Ecpresentations etemp = em.getReference(Ecpresentations.class, showid);
                Ecinfo ef = new Ecinfo();
                ef.setPicy(pics[i]);
                ef.setDetails(comms[i]);
                ef.setShowid(etemp);
                em.persist(ef);
            }
            utx.commit();
            RequestDispatcher view = request.getRequestDispatcher("addinforeport.jsp");
            view.forward(request, response);
            em.close();
        } catch (NotSupportedException ex) {
            ex.printStackTrace();
        } catch (SystemException ex) {
            ex.printStackTrace();
        } catch (RollbackException ex) {
            ex.printStackTrace();
        } catch (HeuristicMixedException ex) {
```

```java
                ex.printStackTrace();
            } catch (HeuristicRollbackException ex) {
                ex.printStackTrace();
            } catch (SecurityException ex) {
                ex.printStackTrace();
            } catch (IllegalStateException ex) {
                ex.printStackTrace();
            }
        }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

Controller/ scheduleshow.java

```java
package Controller;

import Model.Ecperformances;
import Model.Ecpresentations;
import java.io.IOException;
import java.io.PrintWriter;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.List;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.annotation.Resource;
import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.Persistence;
import javax.persistence.PersistenceUnit;
import javax.servlet.RequestDispatcher;
```

```java
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.transaction.HeuristicMixedException;
import javax.transaction.HeuristicRollbackException;
import javax.transaction.NotSupportedException;
import javax.transaction.RollbackException;
import javax.transaction.SystemException;
import javax.transaction.UserTransaction;

/**
 *
 * @author Victor
 */
public class scheduleshow extends HttpServlet {

    @PersistenceUnit(unitName="399A6PU")
    private EntityManagerFactory emf;
    @Resource
    private UserTransaction utx;

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the
left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        RequestDispatcher view = request.getRequestDispatcher("scheduleshow.jsp");
        view.forward(request, response);
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
```

```java
    * @throws ServletException if a servlet-specific error occurs
    * @throws IOException if an I/O error occurs
    */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    try {
            String sid = request.getParameter("showid");
            String title = request.getParameter("title");
            String venue = request.getParameter("ve");
            String genre = request.getParameter("st");
            String sds = request.getParameter("sd");
            String eds = request.getParameter("ed");

            String com = request.getParameter("com");
            String des = request.getParameter("des");

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

            List<String> stime = new ArrayList<String>();
            List<String> sdate = new ArrayList<String>();

            String buff[] = per.split(";");
            for(String substr:buff){
                if(substr == "")
                    break;
                substr = substr.trim();
                String elestr[] = substr.split(",");
                if(elestr.length < 2)
                    break;
                stime.add(elestr[1].trim());
                sdate.add(elestr[0].trim());
            }

            DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
            Date sd = null,ed = null;

            sd = df.parse(sds);
            ed = df.parse(eds);


            Ecpresentations ecpre = new Ecpresentations(sid,venue,sd,ed,genre,title,com,des);

            List<Ecperformances> ecperformancesList = new ArrayList<Ecperformances>();
```

```java
            for(int i = 0;i < stime.size();i ++){
                Ecperformances temp = new Ecperformances();
                temp.setShowid(ecpre);
                temp.setShowtime(stime.get(i));
                temp.setShowdate(df.parse(sdate.get(i)));
                ecperformancesList.add(temp);
            }

            ecpre.setEcperformancesList(ecperformancesList);
            utx.begin();
            EntityManager em = emf.createEntityManager();
            em.persist(ecpre);
            utx.commit();
            em.close();

            request.setAttribute("title", title);
            request.setAttribute("sid", sid);
            RequestDispatcher view = request.getRequestDispatcher("schedulereport.jsp");
            view.forward(request, response);

        } catch (NotSupportedException ex) {
            ex.addSuppressed(ex);
        } catch (SystemException ex) {
            ex.printStackTrace();
        } catch (ParseException ex) {
            ex.printStackTrace();
        } catch (RollbackException ex) {
            ex.printStackTrace();
        } catch (HeuristicMixedException ex) {
            ex.printStackTrace();
        } catch (HeuristicRollbackException ex) {
            ex.printStackTrace();
        } catch (SecurityException ex) {
            ex.printStackTrace();
        } catch (IllegalStateException ex) {
            ex.printStackTrace();
        }
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
```

```java
     */
    @Override
    public String getServletInfo() {
            return "Short description";
    }// </editor-fold>

}
```

```java
package Controller;

import Model.Ecinfo;
import Model.Ecperformances;
import Model.Ecpresentations;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.List;
import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.Persistence;
import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Victor
 */
public class searchreport extends HttpServlet {

    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the
left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
```

```java
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=UTF-8");
        String title = request.getParameter("title");

        EntityManagerFactory emf = Persistence.createEntityManagerFactory("399A6PU");
        EntityManager em = emf.createEntityManager();

        Ecpresentations                          epre                              =
(Ecpresentations)em.createNamedQuery("Ecpresentations.findByTitle").setParameter("title",
title).getSingleResult();
        List<Ecperformances> epers = epre.getEcperformancesList();
        List<Ecinfo> einfo = epre.getEcinfoList();


        request.setAttribute("basic", epre);
        request.setAttribute("time", epers);
        request.setAttribute("sup", einfo);
        RequestDispatcher view = request.getRequestDispatcher("searchreport.jsp");
        view.forward(request, response);
        em.close();
    }

    /**
     * Handles the HTTP <code>POST</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
     * @throws IOException if an I/O error occurs
     */
    @Override
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
```

```java
    }// </editor-fold>

}
```

```java
package Controller;

import Model.Ecpresentations;
import java.io.IOException;
import java.io.PrintWriter;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.List;

import javax.persistence.EntityManager;
import javax.persistence.EntityManagerFactory;
import javax.persistence.Persistence;
import javax.persistence.Query;
import javax.persistence.TemporalType;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletException;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 *
 * @author Victor
 */
public class searchshow extends HttpServlet {


    // <editor-fold defaultstate="collapsed" desc="HttpServlet methods. Click on the + sign on the
left to edit the code.">
    /**
     * Handles the HTTP <code>GET</code> method.
     *
     * @param request servlet request
     * @param response servlet response
     * @throws ServletException if a servlet-specific error occurs
```

```java
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
    RequestDispatcher view = request.getRequestDispatcher("searchshow.jsp");
    view.forward(request, response);
}

/**
 * Handles the HTTP <code>POST</code> method.
 *
 * @param request servlet request
 * @param response servlet response
 * @throws ServletException if a servlet-specific error occurs
 * @throws IOException if an I/O error occurs
 */
@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {

    String st = request.getParameter("st");
    String ve = request.getParameter("ve");
    String sds = request.getParameter("sd");
    String eds = request.getParameter("ed");

    DateFormat df = new SimpleDateFormat("yyyy-MM-dd");
    Date sd = null,ed = null;
    try {
        sd = df.parse(sds);
        ed = df.parse(eds);
    } catch (ParseException ex) {
        ex.printStackTrace();
    }

    EntityManagerFactory emf = Persistence.createEntityManagerFactory("399A6PU");
    EntityManager em = emf.createEntityManager();

    String jpql1 = "SELECT Ecpre FROM Ecpresentations Ecpre WHERE Ecpre.venue = ?1 AND
Ecpre.genre = ?2 AND Ecpre.startseason >= ?3 AND Ecpre.endseason <= ?4";
    Query query = em.createQuery(jpql1);
    query.setParameter(1, ve);
    query.setParameter(2, st);
    query.setParameter(3, sd,TemporalType.DATE);
```

```java
        query.setParameter(4, ed,TemporalType.DATE);

        List<Ecpresentations> result = (List<Ecpresentations>)query.getResultList();
        request.setAttribute("listshow", result);
        request.setAttribute("len", result.size());
        RequestDispatcher view = request.getRequestDispatcher("searchshow.jsp");
        view.forward(request, response);
        em.close();
    }

    /**
     * Returns a short description of the servlet.
     *
     * @return a String containing servlet description
     */
    @Override
    public String getServletInfo() {
        return "Short description";
    }// </editor-fold>

}
```

Model/ Ecinfo.java

```java
package Model;

import java.io.Serializable;
import javax.persistence.Basic;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.Lob;
import javax.persistence.ManyToOne;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
import javax.validation.constraints.NotNull;
import javax.validation.constraints.Size;
import javax.xml.bind.annotation.XmlRootElement;

/**
```

```java
 *
 * @author Victor
 */
@Entity
@Table(name = "ecinfo")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "Ecinfo.findAll", query = "SELECT e FROM Ecinfo e"),
    @NamedQuery(name = "Ecinfo.findByIdECInfo", query = "SELECT e FROM Ecinfo e WHERE
e.idECInfo = :idECInfo")})
public class Ecinfo implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "idECInfo")
    private Integer idECInfo;
    @Basic(optional = false)
    @NotNull
    @Lob
    @Size(min = 1, max = 16777215)
    @Column(name = "details")
    private String details;
    @Lob
    @Size(max = 16777215)
    @Column(name = "picy")
    private String picy;
    @JoinColumn(name = "showid", referencedColumnName = "mykey")
    @ManyToOne(optional = false)
    private Ecpresentations showid;

    public Ecinfo() {
    }

    public Ecinfo(Integer idECInfo) {
        this.idECInfo = idECInfo;
    }

    public Ecinfo(Integer idECInfo, String details) {
        this.idECInfo = idECInfo;
        this.details = details;
    }

    public Integer getIdECInfo() {
```

```java
        return idECInfo;
    }

    public void setIdECInfo(Integer idECInfo) {
        this.idECInfo = idECInfo;
    }

    public String getDetails() {
        return details;
    }

    public void setDetails(String details) {
        this.details = details;
    }

    public String getPicy() {
        return picy;
    }

    public void setPicy(String picy) {
        this.picy = picy;
    }

    public Ecpresentations getShowid() {
        return showid;
    }

    public void setShowid(Ecpresentations showid) {
        this.showid = showid;
    }

    @Override
    public int hashCode() {
        int hash = 0;
        hash += (idECInfo != null ? idECInfo.hashCode() : 0);
        return hash;
    }

    @Override
    public boolean equals(Object object) {
        // TODO: Warning - this method won't work in the case the id fields are not set
        if (!(object instanceof Ecinfo)) {
            return false;
        }
```

```java
        Ecinfo other = (Ecinfo) object;
        if ((this.idECInfo == null && other.idECInfo != null) || (this.idECInfo != null
&& !this.idECInfo.equals(other.idECInfo))) {
            return false;
        }
        return true;
    }

    @Override
    public String toString() {
        return "Model.Ecinfo[ idECInfo=" + idECInfo + " ]";
    }

}
```

Model/ Ecperformances.java

```java
package Model;

import java.io.Serializable;
import java.util.Date;
import javax.persistence.Basic;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.Table;
import javax.persistence.Temporal;
import javax.persistence.TemporalType;
import javax.validation.constraints.NotNull;
import javax.validation.constraints.Size;
import javax.xml.bind.annotation.XmlRootElement;

/**
 *
 * @author Victor
 */
@Entity
```

```java
@Table(name = "ecperformances")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "Ecperformances.findAll", query = "SELECT e FROM Ecperformances e"),
    @NamedQuery(name = "Ecperformances.findByPid", query = "SELECT e FROM Ecperformances e WHERE e.pid = :pid"),
    @NamedQuery(name = "Ecperformances.findByShowtime", query = "SELECT e FROM Ecperformances e WHERE e.showtime = :showtime"),
    @NamedQuery(name = "Ecperformances.findByShowdate", query = "SELECT e FROM Ecperformances e WHERE e.showdate = :showdate")})
public class Ecperformances implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    @Basic(optional = false)
    @Column(name = "pid")
    private Integer pid;
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 8)
    @Column(name = "showtime")
    private String showtime;
    @Basic(optional = false)
    @NotNull
    @Column(name = "showdate")
    @Temporal(TemporalType.DATE)
    private Date showdate;
    @JoinColumn(name = "showid", referencedColumnName = "mykey")
    @ManyToOne(optional = false)
    private Ecpresentations showid;

    public Ecperformances() {
    }

    public Ecperformances(Integer pid) {
        this.pid = pid;
    }

    public Ecperformances(Integer pid, String showtime, Date showdate) {
        this.pid = pid;
        this.showtime = showtime;
        this.showdate = showdate;
    }
```

```java
public Integer getPid() {
    return pid;
}

public void setPid(Integer pid) {
    this.pid = pid;
}

public String getShowtime() {
    return showtime;
}

public void setShowtime(String showtime) {
    this.showtime = showtime;
}

public Date getShowdate() {
    return showdate;
}

public void setShowdate(Date showdate) {
    this.showdate = showdate;
}

public Ecpresentations getShowid() {
    return showid;
}

public void setShowid(Ecpresentations showid) {
    this.showid = showid;
}

@Override
public int hashCode() {
    int hash = 0;
    hash += (pid != null ? pid.hashCode() : 0);
    return hash;
}

@Override
public boolean equals(Object object) {
    // TODO: Warning - this method won't work in the case the id fields are not set
    if (!(object instanceof Ecperformances)) {
```

```java
                return false;
            }
            Ecperformances other = (Ecperformances) object;
            if ((this.pid == null && other.pid != null) || (this.pid != null && !this.pid.equals(other.pid)))
{
                return false;
            }
            return true;
        }

        @Override
        public String toString() {
            return "Model.Ecperformances[ pid=" + pid + " ]";
        }

    }
```

Model/ Ecpresentations

```java
package Model;

import java.io.Serializable;
import java.util.Date;
import java.util.List;
import javax.persistence.Basic;
import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Lob;
import javax.persistence.NamedQueries;
import javax.persistence.NamedQuery;
import javax.persistence.OneToMany;
import javax.persistence.Table;
import javax.persistence.Temporal;
import javax.persistence.TemporalType;
import javax.validation.constraints.NotNull;
import javax.validation.constraints.Size;
import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlTransient;

/**
 *
```

```java
 * @author Victor
 */
@Entity
@Table(name = "ecpresentations")
@XmlRootElement
@NamedQueries({
    @NamedQuery(name = "Ecpresentations.findAll", query = "SELECT e FROM Ecpresentations e"),
    @NamedQuery(name = "Ecpresentations.findByMykey", query = "SELECT e FROM Ecpresentations e WHERE e.mykey = :mykey"),
    @NamedQuery(name = "Ecpresentations.findByVenue", query = "SELECT e FROM Ecpresentations e WHERE e.venue = :venue"),
    @NamedQuery(name = "Ecpresentations.findByStartseason", query = "SELECT e FROM Ecpresentations e WHERE e.startseason = :startseason"),
    @NamedQuery(name = "Ecpresentations.findByEndseason", query = "SELECT e FROM Ecpresentations e WHERE e.endseason = :endseason"),
    @NamedQuery(name = "Ecpresentations.findByGenre", query = "SELECT e FROM Ecpresentations e WHERE e.genre = :genre"),
    @NamedQuery(name = "Ecpresentations.findByTitle", query = "SELECT e FROM Ecpresentations e WHERE e.title = :title"),
    @NamedQuery(name = "Ecpresentations.findByCompany", query = "SELECT e FROM Ecpresentations e WHERE e.company = :company")})
public class Ecpresentations implements Serializable {
    private static final long serialVersionUID = 1L;
    @Id
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 32)
    @Column(name = "mykey")
    private String mykey;
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 9)
    @Column(name = "venue")
    private String venue;
    @Basic(optional = false)
    @NotNull
    @Column(name = "startseason")
    @Temporal(TemporalType.DATE)
    private Date startseason;
    @Basic(optional = false)
    @NotNull
    @Column(name = "endseason")
    @Temporal(TemporalType.DATE)
```

```java
    private Date endseason;
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 18)
    @Column(name = "genre")
    private String genre;
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 128)
    @Column(name = "title")
    private String title;
    @Basic(optional = false)
    @NotNull
    @Size(min = 1, max = 128)
    @Column(name = "company")
    private String company;
    @Basic(optional = false)
    @NotNull
    @Lob
    @Size(min = 1, max = 16777215)
    @Column(name = "shortdescription")
    private String shortdescription;
    @OneToMany(cascade = CascadeType.ALL, mappedBy = "showid")
    private List<Ecperformances> ecperformancesList;
    @OneToMany(cascade = CascadeType.ALL, mappedBy = "showid")
    private List<Ecinfo> ecinfoList;

    public Ecpresentations() {
    }

    public Ecpresentations(String mykey) {
        this.mykey = mykey;
    }

    public Ecpresentations(String mykey, String venue, Date startseason, Date endseason, String
genre, String title, String company, String shortdescription) {
        this.mykey = mykey;
        this.venue = venue;
        this.startseason = startseason;
        this.endseason = endseason;
        this.genre = genre;
        this.title = title;
        this.company = company;
        this.shortdescription = shortdescription;
```

```java
    }

    public String getMykey() {
        return mykey;
    }

    public void setMykey(String mykey) {
        this.mykey = mykey;
    }

    public String getVenue() {
        return venue;
    }

    public void setVenue(String venue) {
        this.venue = venue;
    }

    public Date getStartseason() {
        return startseason;
    }

    public void setStartseason(Date startseason) {
        this.startseason = startseason;
    }

    public Date getEndseason() {
        return endseason;
    }

    public void setEndseason(Date endseason) {
        this.endseason = endseason;
    }

    public String getGenre() {
        return genre;
    }

    public void setGenre(String genre) {
        this.genre = genre;
    }

    public String getTitle() {
        return title;
```

```java
        }

        public void setTitle(String title) {
            this.title = title;
        }

        public String getCompany() {
            return company;
        }

        public void setCompany(String company) {
            this.company = company;
        }

        public String getShortdescription() {
            return shortdescription;
        }

        public void setShortdescription(String shortdescription) {
            this.shortdescription = shortdescription;
        }

        @XmlTransient
        public List<Ecperformances> getEcperformancesList() {
            return ecperformancesList;
        }

        public void setEcperformancesList(List<Ecperformances> ecperformancesList) {
            this.ecperformancesList = ecperformancesList;
        }

        @XmlTransient
        public List<Ecinfo> getEcinfoList() {
            return ecinfoList;
        }

        public void setEcinfoList(List<Ecinfo> ecinfoList) {
            this.ecinfoList = ecinfoList;
        }

        @Override
        public int hashCode() {
            int hash = 0;
            hash += (mykey != null ? mykey.hashCode() : 0);
```

```java
        return hash;
    }


    @Override
    public boolean equals(Object object) {
        // TODO: Warning - this method won't work in the case the id fields are not set
        if (!(object instanceof Ecpresentations)) {
            return false;
        }
        Ecpresentations other = (Ecpresentations) object;
        if ((this.mykey == null && other.mykey != null) || (this.mykey != null
&& !this.mykey.equals(other.mykey))) {
            return false;
        }
        return true;
    }


    @Override
    public String toString() {
        return "Model.Ecpresentations[ mykey=" + mykey + " ]";
    }

}
```

MyScript.js

```javascript
var indexdiv = 0;
function doAdd(){
    var tb = document.getElementById('Myshow');
    var r = tb.insertRow();
    var c1 = r.insertCell();
    var c2 = r.insertCell();

    c1.innerHTML = "<div class = 'targets' id = '" + 'target' + indexdiv + "'></div>";

    var newdev = document.getElementById('target' + indexdiv);
    newdev.addEventListener("dragover",FileDragover,false);
    newdev.addEventListener("drop",FileDrop,false);

    c2.innerHTML = "<textarea rows='7' cols='20' name = 'comms[]'></textarea>"
    indexdiv ++;
}

function handleFileSelect(afile,evt){
```

```javascript
        var reader = new FileReader();
        var filetypes = /^(image\/jpeg|image\/png|image\/gif)$/;

        if(!filetypes.test(afile.type)){
            alert("please select file jpeg png or gif");
            return;
        }

        reader.onload = function(e){
            evt.target.innerHTML += "<img src = '" + e.target.result + "' style='float:left' width ='100'/>";
            evt.target.innerHTML += "<input visibility: hidden name = 'pics[]' value = '" + e.target.result +"'/>";
        }
        reader.readAsDataURL(afile);
}

function FileDragover(evt){
    evt.preventDefault();
}

function FileDrop(evt){
    evt.preventDefault();

    var files = evt.dataTransfer.files;
    var num = files.length;
    var i = 0;
    for(i = 0;i < num;i++){
        var afile = files[i];
        handleFileSelect(afile,evt);
    }
}

function doaddp(){
    var showtime = document.getElementById('stime');
    var date = document.getElementById('d');
    var performances = document.getElementById('Per');

    var sv = showtime.value;
    var dv = date.value;

    var start = document.getElementById('sd');
    var end = document.getElementById('ed');
```

```
var temps = start.value.replace(/-/g,"/");
var tempe = end.value.replace(/-/g,"/");
var tempd = dv.replace(/-/g,"/");

var ds = new Date(Date.parse(temps));
var de = new Date(Date.parse(tempe));
var dd = new Date(Date.parse(tempd));

if(dd >= ds && dd <= de){
    performances.value += dv + ','+ sv +';\n'
}
else{
    alert('the date should be between start and end');
}
}
```