



프로젝트로 배우는 자바 웹 프로그래밍

Servlet, JSP, JDBC

황희정 지음

Chapter 07. JSP와 자바 빈즈

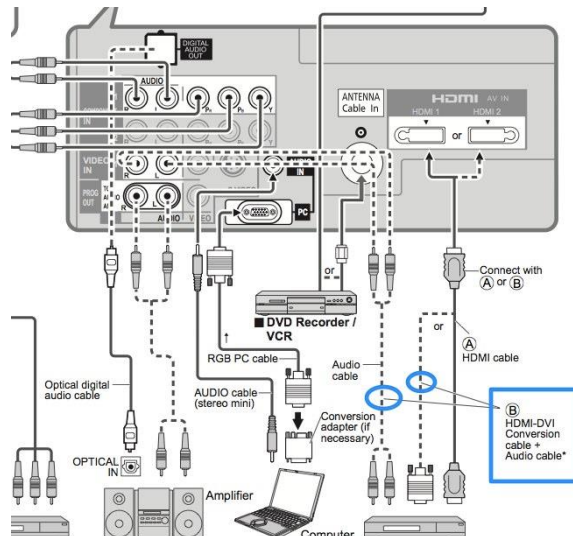
목차

1. 빈즈 개요
2. JSP와 빈즈 연동
3. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현
4. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

01. 빈즈 개요

1. 자바 빈즈

- 일반적으로 컴포넌트라고 하면 다른 무언가를 만들기 위한 부품을 말한다. 컴포넌트는 각각 독립적인 기능이 있으며, 컴포넌트 조합을 통해 다양한 형태의 결과물을 만들 수 있다. 예를 들어, 레고 블록이나 예전의 컴포넌트 오디오 등을 생각해볼 수 있다.
- 이때 각각의 모듈을 서로 조합하려면 규격화된 인터페이스가 있어야 한다. 레고의 경우에는 튀어나온 부분들이 다른 블록의 아래쪽에 결합이 되는 구조고, 컴포넌트 오디오는 일반 스테레오 케이블이나 HDMI, 광케이블 등으로 서로 연결된다.

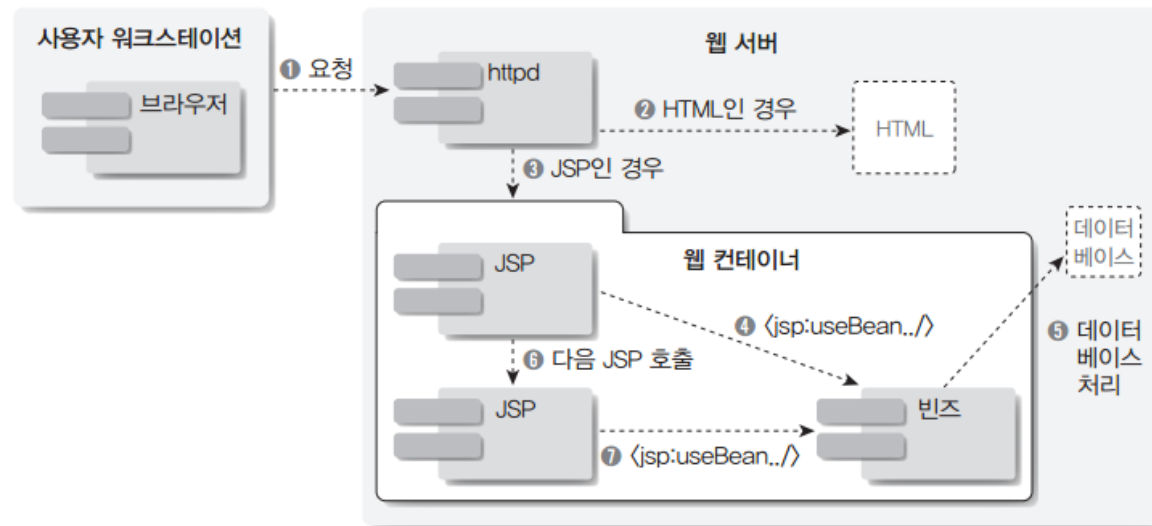


[그림] 레고와 컴포넌트 오디오 인터페이스

01. 빈즈 개요

2. JSP 빈즈

- JSP 빈즈는 **JSP와 연동하기 위해 만들어진 컴포넌트 클래스**이다.
- 컨테이너에 위치하며, JSP에 데이터베이스 연동 등 프로그램적 요소를 모듈화할 수 있도록 도와준다.
- 데이터 처리와 공용화된 기능을 제공하기 때문에 빈즈를 잘 활용하면 프로그램의 중복을 줄이고 더욱 원활하게 유지·보수할 수 있다.
- 따라서 가능하면 JSP 코드 내에 스크립트릿을 사용하는 것보다는 빈즈를 만들어 사용하는 것이 좋다. **(가독성 향상)**

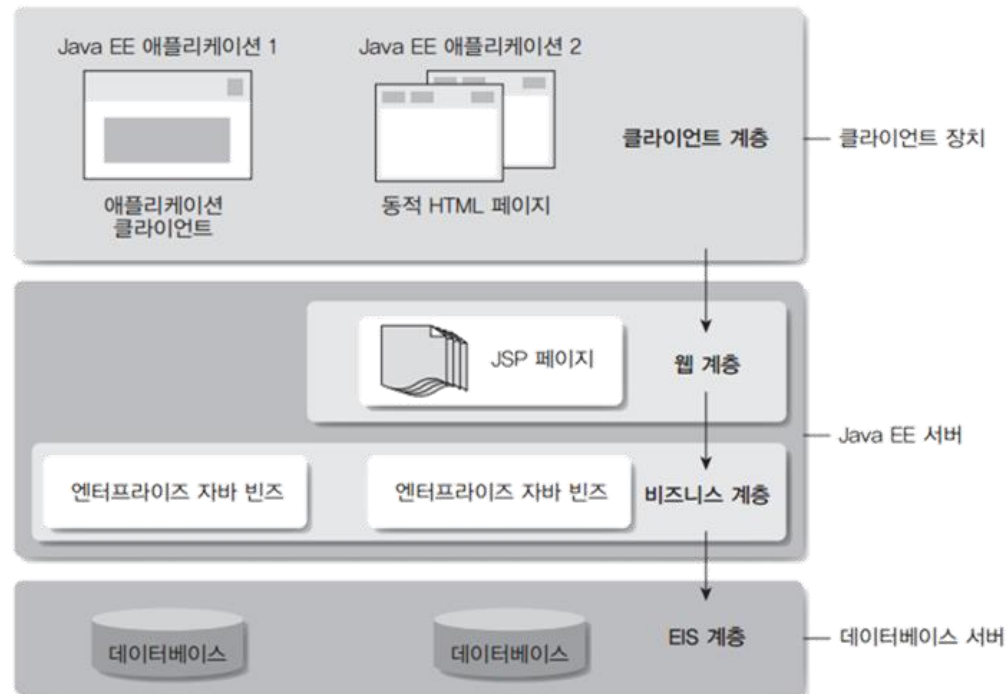


[그림 7-1] 일반적인 JSP 구현에서의 빈즈

01. 빈즈 개요

3. 엔터프라이즈 자바 빈즈

- 엔터프라이즈 자바 빈즈(Enterprise Java Beans, 이하 EJB)는 애플리케이션에서 **비즈니스 로직을 구현하려는 컴포넌트 모델**로, Java EE(Java Enterprise Edition)의 핵심이다.
- Java EE는 **보안을 중시하고 규모가 크며 확장 및 다른 시스템과의 상호·운용을 필요로 하는 애플리케이션** 개발에 필요한 핵심 기술이다. 상당수의 금융기관, 증권사, 이동통신사 등의 업무 시스템들이 Java EE를 기반으로 개발된다.
- 여러 장점에도 불구하고 개발의 복잡성, 컨테이너 간 호환성 등 여러 문제로 인해, 그 대안이 지속적으로 논의 됨.
- JPA(Java Persistence API)를 핵심으로 하는 Java EE 5가 발표되면서(최신 버전은 JavaEE 7) 기존 엔티티빈(EJB 빈 유형 중 하나)의 상당 부분이 대체됨.
- JSP도 JavaEE 의 프레젠테이션 레이어의 한 부분으로 볼 수 있음.



[그림 7-3] 엔터프라이즈 자바 빈즈 구성도

02. JSP와 빈즈 연동

1. 빈즈 연동 개요

- JSP빈즈는 JSP에서 사용할 수 있는 자바 컴포넌트로 빈즈 액션과 결합해 웹 프로그램을 더욱 간편하고 단순한 구조로 개발 할 수 있게 해준다.
- 빈즈도 자바 클래스이기 때문에 JSP 파일의 **스크립트릿 부분에서 인스턴스를 만들어 사용할 수 있다**. 그러나 일반 클래스와 다를 바 없다면 굳이 빈즈라는 이름을 붙일 필요가 없을 것이다.
- 빈즈는 일반 클래스처럼 사용할 수도 있지만, JSP 빈즈만의 특징적인 요소를 잘 활용하면 더욱 편하게 프로그램을 작성할 수 있다.
- 일반적으로 빈즈를 사용할 경우, **사용자 입력 처리에** 필요한 HTML이나 JSP와 폼의 액션에 연결된 JSP 파일이 있어야 한다. 또한 폼을 처리하는 JSP 파일에서 참조할 빈즈 클래스도 필요하다.
- 물론 경우에 따라 **결과를 보여주는 HTML이나 JSP 파일이 필요**할 수도 있다.
- **복잡한 프로그래밍 로직은 자바 빈즈 (i.e., 자바 클래스)와 JSP를 이용해서 처리하고,**
- **사용자 인터페이스 부분은 HTML이나 자바 스크립트를 이용해서 처리**

02. JSP와 빈즈 연동

2. 빈즈 클래스 구조

- 기본적으로 빈즈 클래스는 자바 클래스이므로, 일반적인 자바 클래스 구성을 따른다.
- JSP 액션과 연동하기 위해 필요한 몇 가지 필수 구성의 차이가 있지만 이는 문법적인 제약이 아니기 때문에 규칙을 따르지 않더라도 컴파일 오류가 발생하지는 않는다.
- 빈즈는 **매개변수가 없는 기본 생성자**를 요구하므로 만일 매개변수가 있는 생성자를 구현하였다면 기본 생성자를 명시적으로 선언해 주어야 문제가 발생하지 않는다.
- 빈즈 클래스의 일반적인 구조는 **멤버변수, getter/setter 메소드** 이다.

```
class xxxBean {  
    // 멤버변수 : 데이터베이스 테이블의 칼럼 이름과 매칭된다.  
    private String xxx;  
  
    // get, set 메서드 : 멤버변수와 매칭된다.  
    public String getXxx() {  
        return xxx;  
    }  
    public setXxx(String xxx) {  
        this.xxx = xxx;  
    }  
}
```

02. JSP와 빈즈 연동

3. JSP에서 빈즈 선언

```
<jsp:useBean id="mybean" scope="request" class="MyBean" />
```

[표 7-1] useBean 액션속성 목록

액션	속성	설명
useBean	id	빈즈 클래스의 인스턴스 이름으로 사용할 변수다.
	class	빈즈 클래스의 클래스 이름으로, 패키지 경로를 포함한다.
	scope	빈즈 클래스의 범위로, page, request, session, application이 올 수 있다. (default: page)

- useBean 액션은 다음과 같은 자바 코드로 변경된다. id, class, scope의 의미를 좀 더 자세히 알 수 있다.

```
<%  
MyBean mybean = (MyBean)request.getAttribute("mybean");  
if (mybean == null) {  
    mybean = new MyBean();  
    request.setAttribute("mybean", mybean);  
}  
%>
```

- page scope인 경우

```
<%  
    MyBean mybean = new MyBean();  
%>
```


02. JSP와 빈즈 연동

4. JSP에서 빈즈 속성 설정

- JSP 에서 빈즈를 선언한 다음, `setProperty`, `getProperty` 액션을 통해 빈즈 값을 설정하고 가져올 수 있다.

```
<jsp:setProperty name="mybean" property="userid" />
<jsp:setProperty name="mybean" property="userpasswd" />
혹은
<jsp:setProperty name="mybean" property="*" />
```

[표 7-2] `setProperty` 속성 목록

액션	속성	설명
setProperty	name	빈즈 클래스의 인스턴스 이름으로, id 값에 설정했던 변수 이름이다.
	property	속성 값으로 빈즈 클래스의 <code>setXxx(Xxx는 속성 이름)</code> 메서드와 매칭될 속성 값이다. *를 지정하면 모든 <code>setXxx</code> 에 자동으로 매칭된다.

- 입력항목이 많은 경우 모든 멤버변수에 대해 `setProperty`를 할 필요 없이 `property="*"` 를 사용하면 됨.
- `setProperty`는 다음과 같은 자바 소스 코드로 변환되며 스크립트릿에서도 동일하게 사용할 수 있다.

```
<%
    mybean.setUserId(request.getParameter("userid"));
    mybean.setUserpasswd(request.getParameter("userpasswd"));
%>
```

02. JSP와 빈즈 연동

5. 빈즈 에서 JSP로 속성 데이터 가져오기

- 빈즈에 설정되어 있는 값은 getProperty를 이용해 읽어 올 수 있다. 앞에서 setProperty로 설정된 값은 다음과 같이 JSP에서 가져올 수 있다.

```
<jsp:getProperty name="mybean" property="username" />  
<jsp:getProperty name="mybean" property="userpasswd" />
```

- getProperty 는 화면에 데이터를 출력하는 용도로 사용하기 때문에 HTML과 함께 쓰는 경우가 많다.

```
<TR><TD>이름</TD>  
<TD><jsp:getProperty name="mybean" property="username" /></TD></TR>
```

- 코드가 길어 불편한 경우 <%= %>을 사용할 수도 있다.

```
<TR><TD>ID</TD>  
<TD><%= mybean.getUsername() %></TD></TR>
```

- 스크립트릿 사용

```
<TR><TD>ID</TD>  
<TD>  
<%  
    out.print(mybean.getUsername());  
%></TD></TR>
```

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

■ 실습 개요

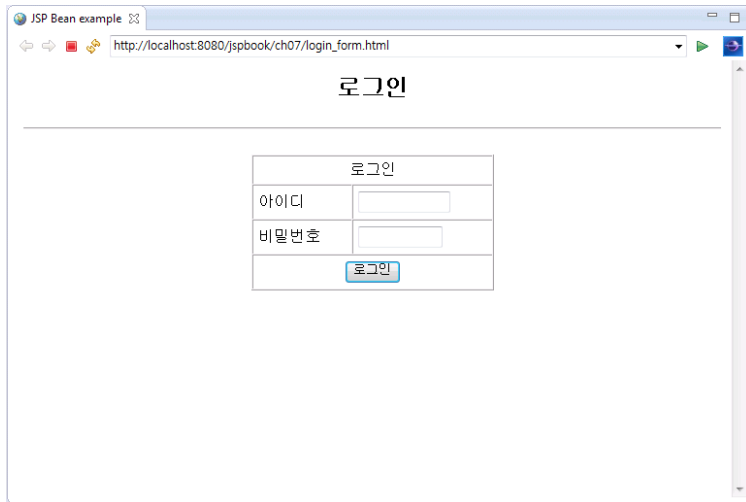
- 웹에서 사용자 로그인을 처리하는 사용자 로그인을 구현하는 간단한 예제를 통해 빈즈 구성과 주변 파일의 구성, 그리고 빈즈를 사용하기 위한 액션 태그 사용을 실습 한다.
- 실제 구현에서는 데이터베이스가 연동되어야 하지만 여기서는 데이터베이스 처리를 가정하고 생성자에서 미리 사용자 인증에 필요한 아이디와 비밀번호를 설정해 둔다.

[표 7-3] 프로그램 소스 목록

파일 이름	역할
login_form.html	사용자 로그인을 위해 아이디와 비밀번호를 입력받는 화면이다.
login.jsp	입력받은 아이디 정보를 빈즈 클래스를 이용해서 확인하고 처리하는 jsp다.
LoginBean.java	사용자가 입력한 계정 정보를 매핑하는 빈즈 클래스로써, 미리 저장된 계정 값과 비교해 로그인 성공 여부를 반환하는 <code>checkUser()</code> 메서드를 포함한다.

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

■ 빈즈 컴파일과 실행



로그인

로그인	
아이디	<input type="text"/>
비밀번호	<input type="password"/>
<input type="button" value="로그인"/>	

[그림 7-9] 입력 화면



로그인 예제

로그인 성공 !!

사용자 아이디 : myuser
사용자 패스워드 : 1234

[그림 7-10] 성공 화면



로그인 예제

로그인 실패 !!

사용자 아이디 : myuser
사용자 패스워드 : 1111

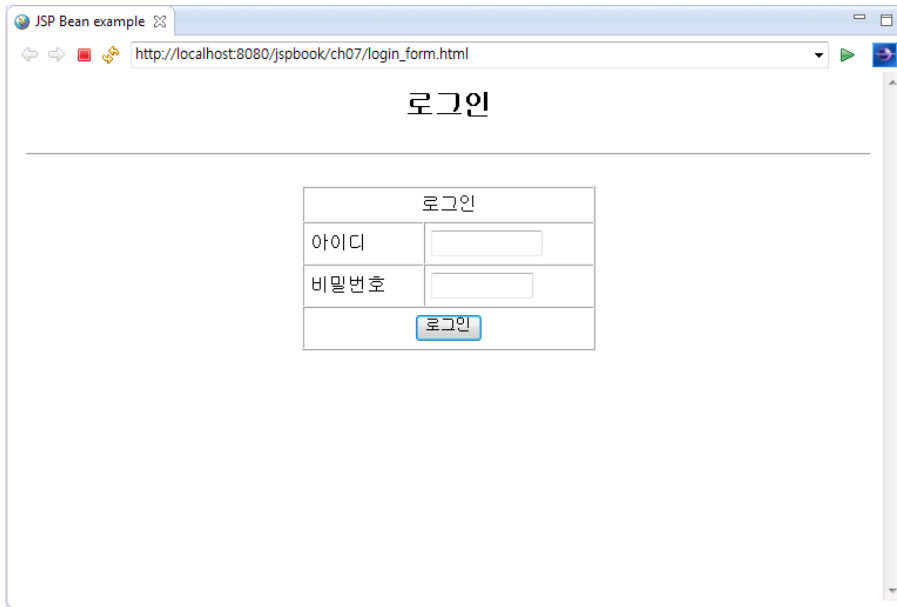
[그림 7-11] 실패 화면

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

1. 사용자 인터페이스 화면 준비

■ [실습] 사용자 정보 입력 화면(login_form.html)

- 교재 p.259 ~ 260 참고



The screenshot shows a web browser window titled "JSP Bean example". The address bar displays "http://localhost:8080/jspbook/ch07/login_form.html". The page content is a login form with the title "로그인" (Login) centered at the top. Below the title is a horizontal line. Underneath the line is a table with the following structure:

로그인	
아이디	<input type="text"/>
비밀번호	<input type="password"/>
<input type="button" value="로그인"/>	

[그림 7-5] 사용자 로그인 양식 화면

[예제 7-1] login_form.html

```
1 <html>
2 <head>
3   <title>ch07 : 로그인</title>
4 </head>
5
6 <body bgcolor="#FFFFFF">
7 <div align=center>
8   <H2>ch07 : 로그인</H2>
9   <HR>
10
11 <form method="post" action="login.jsp" name="form1">
12   <table width="250" border="1" align="center" cellspacing="0" cellpadding="5">
13
14     <tr>
15       <td colspan="2" align="center">로그인</td>
16     </tr>
17     <tr>
18       <td>아이디</td>
19       <td><input type="text" name="userid" size=10></td>
20     </tr>
21     <tr>
22       <td>비밀번호</td>
23       <td><input type="password" name="passwd" size=10></td>
24     </tr>
25     <tr>
26       <td colspan="2" align="center">
27         <input type="submit" name="Submit" value="로그인"></td>
28     </tr>
29   </table>
30 </form>
31 </div>
```

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

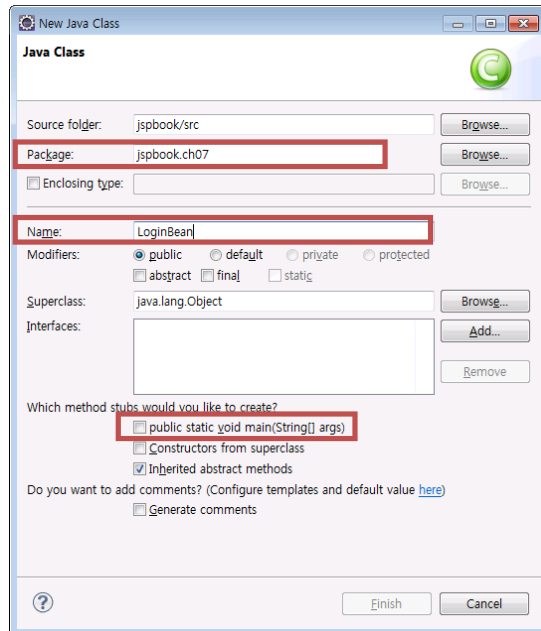
2. 폼 처리를 위한 JSP 파일 준비

■ [실습] 로그인을 처리하는 파일(login.jsp)

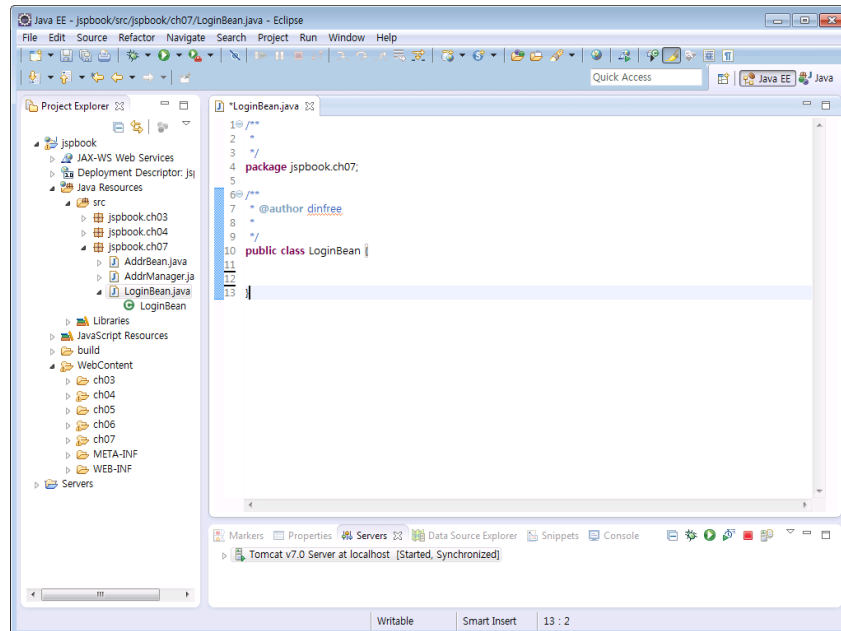
- 교재 p.260 ~ 261참고

3. 사용자 인증을 위한 빈즈 작성

- Package(패키지) : jspbook.ch07
- Name(이름) : LoginBean



[그림 7-6] 클래스 정보 입력



[그림 7-7] 빈즈 클래스 생성

LoginBean.java

```
1 package jspbook.ch07;
2
3 public class LoginBean {
4
5     private String userid;
6     private String passwd;
7
8     final String _userid="myuser";
9     final String _passwd="1234";
10
11     public boolean checkUser() {
12         if(userid.equals(_userid)&&passwd.equals(_passwd)) {
13             return true;
14         }
15         else
16             return false;
17     }
18
19     public String getPasswd() {
20         return passwd;
21     }
22
23     public void setPasswd(String passwd) {
24         this.passwd = passwd;
25     }
26
```

```
27     public String getUserid() {
28         return userid;
29     }
30
31     public void setUserid(String userid) {
32         this.userid = userid;
33     }
34 }
```


[예제 7-2] login.jsp

```
1 <%@ page contentType="text/html; charset=UTF-8" %>
2 <jsp:useBean id="login" class="jspbook.ch07.LoginBean" scope="page" />
3 <jsp:setProperty name="login" property="*" />
4
5 <HTML>
6 <HEAD><TITLE>ch07 : login.jsp </TITLE></HEAD>
7 <BODY>
8 <div align=center>
9 <H2>로그인 예제</H2>
10 <HR>
11 <%
12     if(!login.checkUser()) {
13         out.println("로그인 실패 !!");
14     }
15     else {
16         out.println("로그인 성공 !!");
17     }
18 %>
19
20 <HR>
21 사용자 아이디 : <jsp:getProperty name="login" property="userid" /><BR>
22 사용자 비밀번호 : <jsp:getProperty name="login" property="passwd" />
23
24 </div>
25 </BODY>
26 </HTML>
```

```
if (login.checkUser()) {
    out.println("로그인 성공!!");
}
else {
    out.println("로그인 실패!!");
}
```

04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 실습 개요

- 실제 프로그램 개발에 근접한 응용예제로 주소록 프로그램을 개발한다.
- 데이터베이스는 포함되지 않으며 대신 application scope을 이용해 ArrayList로 데이터를 관리한다.
- DAO, DO 패턴이 적용되었다.

[표 7-4] 프로그램 소스 목록

파일 이름	역할
addr_list.jsp	application scope에 저장된 AddrManager 객체를 통해 저장된 데이터를 가져와서 화면에 출력한다.
addr_form.html	주소록을 등록하기 위한 양식이다.
addr_add.jsp	addr_form.html에서 입력된 내용을 AddrBean으로 매핑하고 AddrManager를 이용해 데이터 추가 기능을 수행한다.
AddrBean.java	주소록 데이터 구조 매핑을 위한 빈즈 클래스다.
AddrManager.java	ArrayList를 이용해 AddrBean 형태의 데이터를 관리하는 클래스다.

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

1. 화면 구성

■ 주소록 목록 화면

[실습] 주소록 목록(addr_list.jsp) - 교재 p.270 ~ 271 참고

■ 주소록 등록 화면과 등록 처리

[실습] 주소록 등록(addr_form.html) - 교재 p.271 ~ 272 참고

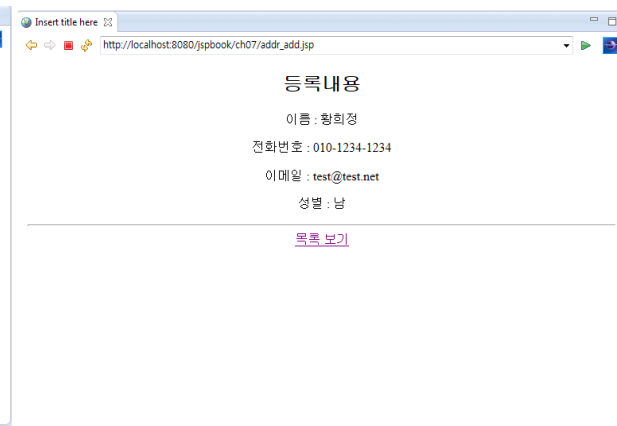
[실습] 주소록 등록 처리(addr_add.jsp) - 교재 p.273 ~ 274 참고



[그림 7-12] 주소록 목록 화면(addr_list.jsp)



[그림 7-13] 주소록 등록 화면(addr_form.html)



[그림 7-14] 주소록 등록 결과 화면(addr_add.jsp)

03. [기본실습] JSP 빈즈 프로그래밍 : 사용자 로그인 구현

2. 빈즈 프로그램의 구현

■ AddrBean 클래스 구현

[실습] 빈즈 클래스로 구현한 주소록(AddrBean.java) - [교재 p.274 ~ 275 참고](#)

■ AddrManager 클래스 구현

[실습] 빈즈 클래스를 이용한 데이터 관리(AddrMamager.java) - [교재 p.276 참고](#)

04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 주요 소스코드 분석(addr_form.html)

```
8 <BODY>
9 <div align="center">
10 <H2>주소록 등록</H2>
11 <HR>
12 <form name=form1 method="post" action=addr_add.jsp>
13 <table border=1 cellspacing="1" cellpadding="5">
14 <tr>
15 <td>이름</td>
16 <td><input type=text size=20 name=username></td>
17 </tr>
18 <tr>
19 <td>전화번호</td>
20 <td><input type=text size=20 name=tel></td>
21 </tr>
22 <tr>
23 <td>이메일</td>
24 <td><input type=text size=20 name=email></td>
25 </tr>
26 <tr>
27 <td>성별</td>
28 <td>
29 <select name=sex >
30 <option selected>남</option>
31 <option>여</option>
32 </select>
33 </td>
34 </tr>
35 <tr><td colspan=2 align=center>
36 <input type=submit value="확인">
37 <input type=reset value="취소"></td></tr>
38 </table>
39 </form>
40 </div>
41 </BODY>
```

04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 주요 소스코드 분석(addr_list.jsp)

```
selProduct.jsp AddrBean.java AddrManager.java addr_add.jsp addr_list.jsp
1 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" import="jspbook.ch07.*"%>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
3 <jsp:useBean id="am" class="jspbook.ch07.AddrManager" scope="application"/>
4
5 <html>
6 <head>
7 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
8 <title>ch07 : 주소록 목록</title>
9 </head>
10 <body>
11 <div align=center>
12 <H2>주소록</H2>
13 <HR>
14 <a href="addr_form.html">주소추가</a><P>
15 <table border=1 width=500>
16 <tr><td>이름</td><td>전화번호</td><td>이메일</td><td>성별</td></tr>
17 <%
18     for(AddrBean ab : am.getAddrList()) {
19 %>
20 <tr>
21 <td><%=ab.getUsername() %></td>
22 <td><%=ab.getTel() %></td>
23 <td><%=ab.getEmail() %></td>
24 <td><%=ab.getSex() %></td></tr>
25 <%
26     }
27 %>
28 </table>
29 </div>
30 </body>
31 </html>
```

04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 주요소스코드 분석(addr_add.jsp)

```
ProductList.jsp  logtest.jsp  HelloWorld.jsp  ItemTagTest.jsp  testform.html  add.jsp  addr_add.jsp  >>
1  <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" import="jspbook.ch07.*"%>
2  <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
3  <% request.setCharacterEncoding("UTF-8"); %>
4
5  <jsp:useBean id="addr" class="jspbook.ch07.AddrBean"/>
6  <jsp:setProperty name="addr" property="*" />
7  <jsp:useBean id="am" class="jspbook.ch07.AddrManager" scope="application"/>
8  <%
9      am.add(addr);
10 %>
11 <html>
12 <head>
13 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
14 <title>ch07 : addr_add.jsp</title>
15 </head>
16 <body>
17 <div align="center">
18 <H2>등록내용</H2>
19 이름 : <jsp:getProperty property="username" name="addr"/><P>
20 전화번호 : <%=addr.getTel() %><P>
21 이메일 : <%=addr.getEmail() %> <P>
22 성별 : <%=addr.getSex() %>
23 <HR>
24 <a href="addr_list.jsp">목록 보기</a>
25 </div>
26 </body>
27 </html>
28
```

04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 주요 소스코드 분석(AddrBean.java)

```
seirProduct.jsp AddrBean.java x
1 package jspbook.ch07;
2
3 public class AddrBean {
4     private String username;
5     private String tel;
6     private String email;
7     private String sex;
8
9     public String getUsername() {
10         return username;
11     }
12     public void setUsername(String username) {
13         this.username = username;
14     }
15     public String getTel() {
16         return tel;
17     }
18     public void setTel(String tel) {
19         this.tel = tel;
20     }
21     public String getEmail() {
22         return email;
23     }
24     public void setEmail(String email) {
25         this.email = email;
26     }
27     public String getSex() {
28         return sex;
29     }
30     public void setSex(String sex) {
31         this.sex = sex;
32     }
33 }
34
```


04. [응용실습] JSP 빈즈 프로그래밍 : 주소록 구현

■ 주요 소스코드 분석(AddrManager.java)

```
1 package jspbook.ch07;
2
3 import java.util.ArrayList;
4 import java.util.List;
5
6 public class AddrManager {
7     List<AddrBean> addrlist = new ArrayList<AddrBean>();
8
9     public void add(AddrBean ab) {
10         addrlist.add(ab);
11     }
12
13     public List<AddrBean> getAddrList() {
14         return addrlist;
15     }
16
17     public AddrBean getAddr(String username) {
18         for(AddrBean ab : addrlist) {
19             if(ab.getUsername().equals(username))
20                 return ab;
21         }
22         return null;
23     }
24 }
25
```



프로젝트로 배우는 자바 웹 프로그래밍

Servlet, JSP, JDBC

황희정 지음