



자바빈과 <jsp:useBean> 액션 태그

Contents

- ❖ 1. 자바빈(JavaBeans)의 개요
- ❖ 2. 자바빈 클래스 만들기
- ❖ 3. <jsp:useBean> 액션 태그를 이용한 자바 객체 사용

페이지 간 데이터 전송

❖ 입력 폼을 이용한 데이터 전송의 경우

<loginForm.jsp>

A screenshot of a web browser window showing the URL `localhost:8181/week5/loginForm.jsp`. The page title is ★ 로그인 화면 ★. It contains two input fields: '아이디' (ID) with value 'sunyae' and '비밀번호' (Password) with value '....'. Below the fields is a '로그인' (Login) button.

<loginProcess.jsp>

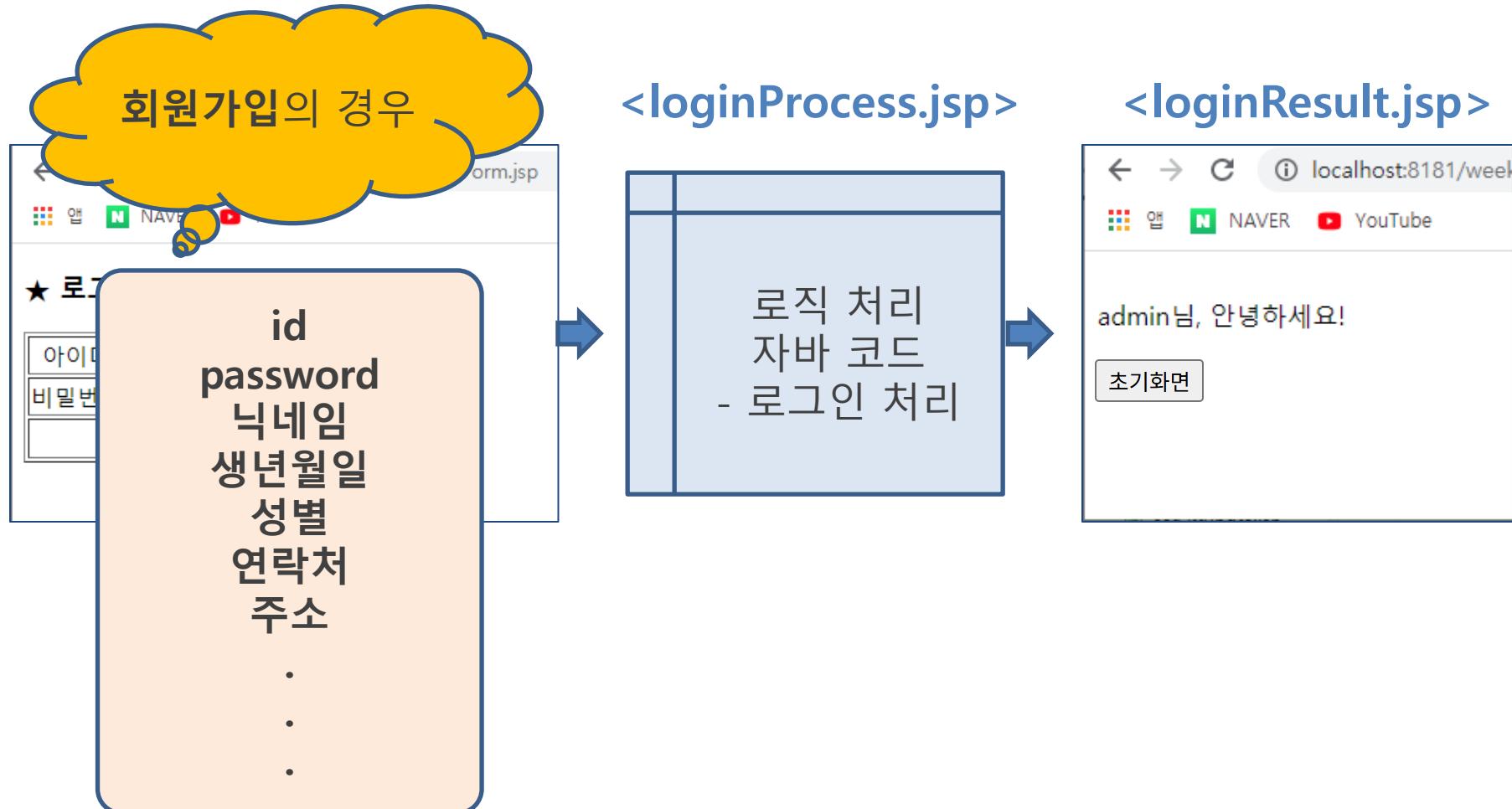


<loginResult.jsp>

A screenshot of a web browser window showing the URL `localhost:8181/week5/loginResult.jsp`. The page displays the message "admin님, 안녕하세요!" (Hello admin!) and a "초기화면" (Initial Screen) button.

페이지 간 데이터 전송

❖ 입력 폼을 이용한 데이터 전송의 경우



자바빈(JavaBean)

❖ 자바빈 이란?

- JSP/서블릿에서 데이터를 저장하고 전달하기 위해 사용하는 재사용 가능한 자바 클래스
- JSP의 <jsp:useBean> 액션 태그로 쉽게 사용 가능
- MVC 패턴에서 주로 **Model(데이터 역할)**로 사용
 - DTO(Data Transfer Object) 역할

자바빈(JavaBean)

❖ 자바빈 규약(작성 규칙)

항 목	설 명
기본 생성자	public 생성자 필수
필드	private 접근제한자 사용
메서드	각 필드마다 public getXXX()/setXXX() 메서드 작성
직렬화	필요 시 implement Serializable 구현

자바빈(JavaBean)

❖ 자바빈 규약(작성 규칙)

항 목	설 명
기본 생성자	public 생성자 필수
필드	private 접근제한자 사용
메서드	각 필드마다 public getXXX()/setXXX() 메서드 작성
직렬화	필요 시 implement Serializable 구현

JVM 메모리의 객체 데이터를
바이트 형태로 변환하는 기술

자바빈(JavaBean)의 작성

- ❖ 자바빈은 자바 클래스이므로 자바 클래스를 만드는 것과 동일한 규칙을 따름
- ❖ 작성
 - package 명
 - package 패키지명;
 - package week5;
 - 해당 클래스를 생성할 때 필요한 클래스명(패키지명을 포함)
 - import 패키지명.클래스명;
 - import java.util.Date;
 - class 문
 - public class 클래스명;

자바빈(JavaBean)의 작성

❖ 자바빈의 속성(property, 멤버변수) 선언

- property(프로퍼티) 는 값을 저장하기 위한 필드로 접근제어자를 private로 선언해서 작성

- **private String name;**
- **private String userId;**

❖ 자바빈의 메소드 선언

- property에 접근하기 위한 getter(), setter() 메소드는 접근제어자를 public으로 선언해서 작성

- **public void setId(String userId) {
 this.userId = userId; }**
- **public String getId() {
 return userId; }**

자바빈(JavaBean)의 작성

❖ 자바빈의 속성(property, 멤버변수) 선언

- property(프로퍼티)는 값을 저장하기 위한 필드로 접근제어자를 private로 선언해서 작성

- **private String name;**
- **private String userId;**

❖ 자바빈의 메소드 선언

- property에 접근하기 위한 getter(), setter() 메소드는 접근제어자를 public으로 선언해서 작성

- **public void setId(String userId) {
 this.userId = userId; }**
- **public String getId() {
 return userId; }**

데이터 저장소인 property에
값을 저장할 때 사용

property에 저장된
값을 읽어올 때 사용

자바빈(JavaBean) 클래스 작성

- ❖ 예제로 사용할 자바빈 클래스 작성
- ❖ 회원 정보를 저장할 때 사용할 클래스
- ❖ MemberInfo 자바빈의 프로퍼티 목록

프로퍼티 이름	값 타입	설 명
id	String	회원 아이디
password	String	회원 암호
name	String	회원 이름
email	String	회원 이메일
registerDate	String	회원 등록일자

자바빈(JavaBean) 클래스 작성

- ❖ week5 > Java Resources > src > week5 > MemberInfo.java
- ❖ package > week5

```
package week5;

public class MemberInfo {
    private String id;
    private String password;
    private String name;
    private String email;
    private String registerDate;
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
}

public String getPassword() {
    return password;
}
public void setPassword(String password) {
    this.password = password;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}
public String getRegisterDate() {
    return registerDate;
}
public void setRegisterDate(String registerDate) {
    this.registerDate = registerDate;
}
```

자바빈(JavaBean) 클래스 작성

- ❖ week5 > Java Resources > src > week5 > MemberInfo.java
- ❖ package > week5

```
package week5;

public class MemberInfo {
    private String id;
    private String password;
    private String name;
    private String email;
    private String registerDate;
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
}
```

getter()/setter() 자동 생성

source > Generate gatter/setter >
select all > generate

```
        public String getPassword() {
            return password;
        }
        public void setPassword(String password) {
            this.password = password;
        }
        public String getName() {
            return name;
        }
        public void setName(String name) {
            this.name = name;
        }
        public String getEmail() {
            return email;
        }
        public void setEmail(String email) {
            this.email = email;
        }
        public String getRegisterDate() {
            return registerDate;
        }
        public void setRegisterDate(String registerDate) {
            this.registerDate = registerDate;
        }
    }
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

- ❖ JSP 페이지의 주된 기능은 데이터를 보여주는 것
 - 게시판의 글 목록보기, 글 읽기, 회원 정보 보기 등
- ❖ 자바빈과 같은 클래스에 담아서 값을 보여주는 것이 일반적
- ❖ 자바빈을 사용하기 위한 3가지 액션태그

자바빈 액션 태그	설명
<jsp:useBean>	자바빈 객체를 생성하는 액션 태그
<jsp:setProperty>	생성된 자바빈 객체에 프로퍼티 값을 저장하는 액션 태그
<jsp:getProperty>	생성된 자바빈 객체에서 저장된 프로퍼티 값을 가져오는 액션 태그

<jsp:useBean> 태그를 이용한 자바 객체 사용

- ❖ <jsp:useBean> 액션 태그를 사용하여 객체 생성하기

```
<jsp:useBean id="[빈 이름]" class="[자바빈클래스이름]" scope="[범위]" />
```

액션 태그 속성	설명
id	JSP 페이지에서 자바빈 객체에 접근할 때 사용할 이름을 지정
class	패키지 이름을 포함한 자바빈 클래스의 완전한 이름을 입력
scope	자바빈 객체를 저장할 영역을 지정 page, request, session, application 중 하나를 값으로 가진다. 기본값은 page이다.

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:useBean> 액션 태그를 사용하여 객체 생성하기

```
<jsp:useBean id="[빈 이름]" class="[자바빈클래스이름]" scope="[범위]" />
```

액션 태그 속성	설명
id	JSP 페이지에서 자바빈 객체에 접근할 때 사용할 이름을 지정
class	패키지 이름을 포함한 자바빈 클래스의 완전한 이름을 입력
scope	자바빈 객체를 저장할 영역을 지정 page, request, session, application 중 하나를 값으로 가진다. 기본값은 page이다.

❖ 사용 예

```
<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:setProperty> 액션 태그

- 생성한 자바빈 객체의 프로퍼티 값 변경 가능

```
<jsp:setProperty name="[자바빈]" property="이름" value="[값]" />
```

액션 태그 속성	설 명
name	프로퍼티의 값을 변경할 자바빈 객체의 이름을 지정한다. <jsp:useBean> 액션 태그의 id 속성에서 지정한 값을 사용한다.
property	값을 지정할 프로퍼티의 이름을 지정한다.
value	프로퍼티의 값을 지정한다. 표현식(<%= 값%>)이나 EL(\${값})을 사용할 수 있다.

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:setProperty> 액션 태그

- 생성한 자바빈 객체의 프로퍼티 값 변경 가능

```
<jsp:setProperty name="[자바빈]" property="이름" value="[값]" />
```

액션 태그 속성	설명
name	프로퍼티의 값을 변경할 자바빈 객체의 이름을 지정한다. <jsp:useBean> 액션 태그의 id 속성에서 지정한 값을 사용한다.
property	값을 지정할 프로퍼티의 이름을 지정한다.
value	프로퍼티의 값을 지정한다. 표현식(<%= 값%>)이나 EL(\${값})을 사용할 수 있다.

- 사용 예

```
<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
```

```
<jsp:setProperty name="member" property="name" value="홍길동" />
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:getProperty> 액션 태그

- 생성한 자바빈 객체에 저장된 프로퍼티 값을 읽기

```
<jsp:getProperty name="[자바빈]" property="프로퍼티이름" />
```

액션 태그 속성	설명
name	<jsp:useBean> 액션 태그의 id 속성에서 지정한 값을 사용 한다.
property	출력할(값을 읽어올) 프로퍼티의 이름을 지정한다.

<jsp:useBean> 태그를 이용한 자바 객체 사용

- ❖ <jsp:getProperty> 액션 태그
 - 생성한 자바빈 객체에 저장된 프로퍼티 값을 읽어올 수 있다.

```
<jsp:getProperty name="[자바빈]" property="프로퍼티이름" />
```

액션 태그 속성	설명
name	<jsp:useBean> 액션 태그의 id 속성에서 지정한 값을 사용 한다.
property	출력할(값을 읽어올) 프로퍼티의 이름을 지정한다.

- 사용 예

```
<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
```

```
<jsp:getProperty name="member" property="name" />
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:useBean> 액션 태그를 사용하는 예제 작성

- MemberInfo 객체를 생성
- <jsp:forward>를 사용하여 다른 페이지로 이동

❖ makeObject.jsp

```
<body>

<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
<jsp:setProperty property="id" name="member" value="admin"/>
<jsp:setProperty property="name" name="member" value="홍길동"/>
<jsp:setProperty property="email" name="member" value="hgd@naver.com"/>

<jsp:forward page="useObject.jsp"></jsp:forward>

</body>
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:useBean> 액션 태그를 사용하는 예제 작성

- MemberInfo 객체를 생성
- <jsp:forward>를 사용하여 다른 페이지로 이동

❖ useObject.jsp

```
<body>
```

```
    <jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
```

```
    이 름 : <jsp:getProperty property="name" name="member"/>
```

```
    (<jsp:getProperty property="id" name="member"/>) 님, 안녕하세요.
```

```
    <br>
```

```
    당신의 이메일은 <jsp:getProperty property="email" name="member"/>입니다.
```

```
</body>
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

- ❖ <jsp:useBean> 액션 태그를 사용하는 예제 작성
 - MemberInfo 객체를 생성
 - <jsp:forward>를 사용하여 다른 페이지로 이동

❖ makeObject.

request.setAttribute("member", MemberInfo객체); 의미

```
<body>
```

```
<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
<jsp:setProperty property="id" name="member" value="admin"/>
<jsp:setProperty property="name" name="member" value="홍길동"/>
<jsp:setProperty property="email" name="member" value="hgd@naver.com"/>
```

```
<jsp:forward page="useObject.jsp"></jsp:forward>
```

```
</body>
```

<jsp:useBean> 태그를 이용한 자바 객체 사용

❖ <jsp:useBean> 액션 태그를 사용하는 예제 작성

- MemberInfo 객체를 생성
- <jsp:forward>를 사용하여 다른 페이지로 이동

❖ useObject2.jsp

```
<body>

<%
    MemberInfo member = (MemberInfo)request.getAttribute("member");
%>

이 름 : <%=member.getName() %> (<%=member.getId() %>)님, 안녕하세요.
<br>
당신의 이메일은 <%=member.getEmail() %>입니다.

</body>
```

자바빈을 이용한 회원가입

❖ 회원가입 입력 폼 만들기(memberForm.jsp)

* 회원가입 *

아이디	<input type="text"/>
비밀번호	<input type="password"/>
이름	<input type="text"/>
이메일	<input type="text"/>
<input type="button" value="회원가입"/>	

자바 웹 애플리케이션 개발

```
<body>
    <h2>* 회원가입 *</h2>
    <form action="memberJoin.jsp">
        <table border="1">
            <tr>
                <td>아이디 </td>
                <td><input type="text" name="id"> </td>
            </tr>
            <tr>
                <td>비밀번호 </td>
                <td><input type="password" name="password"> </td>
            </tr>
            <tr>
                <td>이름 </td>
                <td><input type="text" name="name"> </td>
            </tr>
            <tr>
                <td>이메일 </td>
                <td><input type="text" name="email"> </td>
            </tr>
            <tr>
                <td colspan="2" align="center">
                    <input type="submit" value="회원가입">
                </td>
            </tr>
        </table>
    </form>
</body>
```

자바빈을 이용한 회원가입

❖ 회원가입 처리

<memberForm.jsp>

* 회원가입 *

아이디	<input type="text"/>
비밀번호	<input type="password"/>
이름	<input type="text"/>
이메일	<input type="text"/>
<input type="button" value="회원가입"/>	

<memberJoin.jsp>

로직 처리
자바 코드
- 회원가입 처리

<memberResult.jsp>

* 회원가입 결과 *

아이디	admin
비밀번호	1234
이름	홍길동
이메일	hgd@naver.com
가입일자	2025-09-21

초기화면

자바빈을 이용한 회원가입

❖ 회원가입 처리(memberJoin.jsp)

```
<body>
<%
    Date date = new Date();
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
    String strDate = sdf.format(date);
%>

<jsp:useBean id="member" class="week5.MemberInfo" scope="request" />
<jsp:setProperty property="*" name="member"/>
<jsp:setProperty property="registerDate" name="member" value="<%="strDate %>" />

<jsp:forward page="memberResult.jsp"></jsp:forward>

</body>
```

자바빈을 이용한 회원가입

❖ 회원가입 결과(memberResult.jsp)

```
<body>
<%
    MemberInfo member = (MemberInfo)request.getAttribute("member");
%>
<h2>* 회원가입 결과 *</h2>
<table border="1" cellpadding="5">
    <tr>
        <td>아이디 </td>
        <td><%=member.getId() %> </td>
    </tr>
    <tr>
        <td>비밀번호 </td>
        <td><%=member.getPassword() %> </td>
    </tr>
    <tr>
        <td>이름 </td>
        <td><%=member.getName() %> </td>
    </tr>
    <tr>
        <td>이메일 </td>
        <td><%=member.getEmail() %> </td>
    </tr>
    <tr>
        <td>가입일자 </td>
        <td><%=member.getRegisterDate() %> </td>
    </tr>
</table>
<br><br>
<button onclick="location.href='memberForm.jsp'">초기화면</button>
</body>
```

실습 문제

❖ Book 클래스를 이용한 데이터 저장 및 조회

프로퍼티 이름	값 타입	설명
code	String	코드
title	String	제목
writer	String	저자
pubDate	String	출판일자

◆ 도서 등록 화면 ◆

도서 코드	<input type="text"/>
도서 제목	<input type="text"/>
도서 저자	<input type="text"/>
<input type="button" value="등록"/>	

◆ 도서 등록 결과 ◆

코드	제목	저자	출판일자
A0001	자바빈 이용 방법	작가 J.	2019-10-13

실습 문제

❖ Book 클래스를 이용한 데이터 저장 및 조회

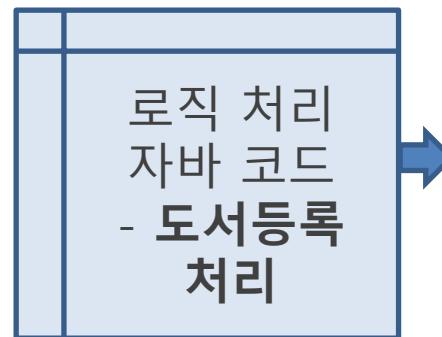
프로퍼티 이름	값 타입	설명
code	String	코드
title	String	제목
writer	String	저자
pubDate	String	출판일자

<bookForm.jsp>

◆ 도서 등록 화면 ◆

도서 코드	<input type="text"/>
도서 제목	<input type="text"/>
도서 저자	<input type="text"/>
<input type="button" value="등록"/>	

<bookJoin.jsp>



<bookResult.jsp>

◆ 도서 등록 결과 ◆

코드	제목	저자	출판일자
A0001	자바빈 이용 방법	작가 J.	2019-10-13

실습 문제

❖ Book 클래스를 이용한 데이터 저장 및 조회

```
2
3 public class Book {
4     private String code;
5     private String title;
6     private String writer;
7     private String pubDate;
8     public String getCode() {
9         return code;
10    }
11    public void setCode(String code) {
12        this.code = code;
13    }
14    public String getTitle() {
15        return title;
16    }
17    public void setTitle(String title) {
18        this.title = title;
19    }
20    public String getWriter() {
21        return writer;
22    }
23    public void setWriter(String writer) {
24        this.writer = writer;
25    }
26    public String getPubDate() {
27        return pubDate;
28    }
29    public void setPubDate(String pubDate) {
30        this.pubDate = pubDate;
31    }
32 }
```

실습 문제

도서 등록

도서 코드

P1234

도서 제목

쉽게 배우는 JSP

도서 저자

홍길동

등록

도서 등록 결과

제 목	내 용
도서코드	P1234
도서제목	쉽게 배우는 JSP
도서저자	홍길동
출판일자	2022-10-03

실습 문제

❖ bookForm.jsp

bootstrap 사용하기

```
<title>Insert title here</title>
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/css/bootstrap.min.css" rel="stylesheet">
</head>
```

```
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.8/dist/js/bootstrap.bundle.min.js">
</body>
```

부트스트랩 사이트 > 문서 > 시작하기 > css, js 포함하기 위해
link, script 태그 복사

실습 문제

❖ bookForm.jsp

```
<body>
<div class="container">
    <form action="bookJoin.jsp">
        <div style="margin: 20px; padding: 20px; background-color: #f2f1ed;">
            <h1 class="display-1">도서 등록</h1>
        </div>
        <div style="margin: 10px; padding: 20px;">
            <div class="mb-3">
                <label for="exampleFormControlInput1" class="form-label">도서코드</label>
                <input type="text" name="code" class="form-control" id="exampleFormControlInput1" >
            </div>
            <div class="mb-3">
                <label for="exampleFormControlInput1" class="form-label">도서제목</label>
                <input type="text" name="title" class="form-control" id="exampleFormControlInput1" >
            </div>
            <div class="mb-3">
                <label for="exampleFormControlInput1" class="form-label">도서저자</label>
                <input type="text" name="writer" class="form-control" id="exampleFormControlInput1" >
            </div>
            <div class="mb-3">
                <input type="submit" value="등록" class="btn btn-primary" >
            </div>
        </div>
    </form>
</div>
```

실습 문제

❖ bookJoin.jsp

```
<body>
<%
    Date date = new Date();
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
    String strDate = sdf.format(date);
%>

<jsp:useBean id="book" class="week5.Book" scope="request" />
<jsp:setProperty property="*" name="book"/>
<jsp:setProperty property="pubDate" name="book" value="<%="strDate %>" />

<jsp:forward page="bookResult.jsp" />

</body>
```

실습 문제

bootstrap 사용하기 : link, script 태그 포함

❖ bookResult.jsp

```
<body>
<%
    Book book = (Book)request.getAttribute("book");
%>
<div class="container">
    <div style="margin: 20px; padding: 20px; background-color: #f2f1ed;">
        <h1 class="display-1">도서 등록 결과</h1>
    </div>
    <div style="margin: 10px; padding: 20px;">
        <table class="table table-striped">
            <thead>
                <tr>
                    <th scope="col">제 목</th>
                    <th scope="col">내 용</th>
                </tr>
            </thead>
```

실습 문제

bootstrap 사용하기 : link, script 태그 포함

❖ bookResult.jsp

```
<tbody>
    <tr>
        <td>도서코드</td>
        <td><%=book.getCode() %></td>
    </tr>
    <tr>
        <td>도서제목</td>
        <td><%=book.getTitle() %></td>
    </tr>
    <tr>
        <td>도서저자</td>
        <td><%=book.getWriter() %></td>
    </tr>
    <tr>
        <td>출판일자</td>
        <td><%=book.getPubDate() %></td>
    </tr>
</tbody>
</table>

<div class="mb-3">
    <button onclick="location.href='bookForm.jsp'" class="btn btn-primary" >도서등록</button>
</div>
</div>
</div>
```

실습 문제

도서 등록

도서 코드

P1234

도서 제목

쉽게 배우는 JSP

도서 저자

홍길동

등록

도서 등록 결과

제 목	내 용
도서코드	P1234
도서제목	쉽게 배우는 JSP
도서저자	홍길동
출판일자	2022-10-03

Bootstrap

❖ Bootstrap

- 2010년 중반에 트위터 개발자 중 트위터 아이디 @mdo와 @fat을 사용하는 개발자가 만든 오픈소스 프레임워크
 - HTML, CSS, JavaScript 및 다양한 UI 컴포넌트로 구성
 - 장점 - 부트스트랩만 잘 활용해도 웹 사이트를 빠르고 간편하게 개발 가능
-
- 사이트
 - <https://getbootstrap.kr>

Bootstrap

■ 부트스트랩

v5.3 업데이트 색상 테마 지원, 확장된 색상 팔레트 등이 추가되었습니다!

Bootstrap으로 빠르게 반응형 사이트 빌드하기

강력하고 확장 가능하며 기능이 풍부한 프론트엔드 툴킷. Sass로 빌드 및 커스터마이징하고, 사전 빌드된 그리드 시스템 및 구성 요소를 활용하고, 강력한 JavaScript 플러그인으로 프로젝트에 생기를 불어넣습니다.

\$ npm i bootstrap@5.3.3

문서 읽어보기

현재 v5.3.3 · 다운로드 · 모드 버저

Bootstrap

■ 부트스트랩 > 문서 > 컨텐츠 / 폼 / 컴포넌트 등

컨텐츠

Reboot

타이포그래피

이미지

테이블

피규어

폼

개요

폼 컨트롤

셀렉트

체크박스 & 라디오버튼

범위

입력 그룹

플로팅 라벨

레이아웃

유효성 검사

컴포넌트

아코디언

경고창

배지

브레드크럼

버튼

버튼 그룹

카드

캐러셀

닫기 버튼

콜랩스

드롭다운

목록 그룹

모달

내비게이션 바

내비게이션 & 탭

Bootstrap

■ 부트스트랩 > 문서 > 컨텐츠 / 폼 / 컴포넌트 등

컨텐츠

Reboot

타이포그래피

이미지

테이블

피규어

폼

개요

폼 컨트롤

셀렉트

체크박스 & 라디오버튼

범위

입력 그룹

플로팅 라벨

레이아웃

유효성 검사

컴포넌트

아코디언

경고창

배지

브레드크럼

버튼

버튼 그룹

카드

캐러셀

닫기 버튼

콜랩스

드롭다운

목록 그룹

모달

내비게이션 바

내비게이션 & 탭

과제물

❖ 개인 정보 입력 및 결과 화면 만들기

< 개인정보 입력 화면 >

1. 이름 :
2. 나이 :
3. 성별 : 여자 남자
4. 주소 :
5. 좋아하는 운동
 - 농구
 - 축구
 - 탁구
 - 헬스

입력

< 개인정보 입력 결과 >

구분	내용
이름	홍길동
나이	25세
성별	남자
주소	인천시 미추홀구 xxx동
좋아하는 운동	탁구, 헬스

과제물

❖ 개인 정보 입력 및 결과 화면 만들기

개인 정보 입력

이름

나이

성별 남자 여자

주소

좋아하는 운동 농구 축구 탁구 헬스

등록

개인 정보 결과

구 분

내 용

이 름

홍길동

나 이

25세

성 별

남자

주 소

서울

좋아하는 운동

농구 축구 탁구 헬스

초기화면

❖ 개인 정보 입력 및 결과 화면 만들기

❖ 제출방법

- 프로젝트명(Dynamic Web Project) : **hw3_학번**
- 클래스 Person
 - private String name
 - private int age
 - private String gender
 - private String addr
 - private String[] sports
- personForm.jsp
- personProcess.jsp
- personResult.jsp

- 프로젝트명 오른쪽 마우스 클릭 > Export > War file 선택
- 최종 생성된 **hw3_학번.war** 파일을 제출