

# JSP TO MVC 프로젝트

멀티 플랫폼 서비스

중고 물품 커뮤니티

# 목 차

---

프로젝트 개요

팀 구성원 역할

기존 프로젝트 설명

MVC 프로젝트 변경 수행 절차

MVC 프로젝트로의 변경점

자체 평가

# 프로젝트 개요

---

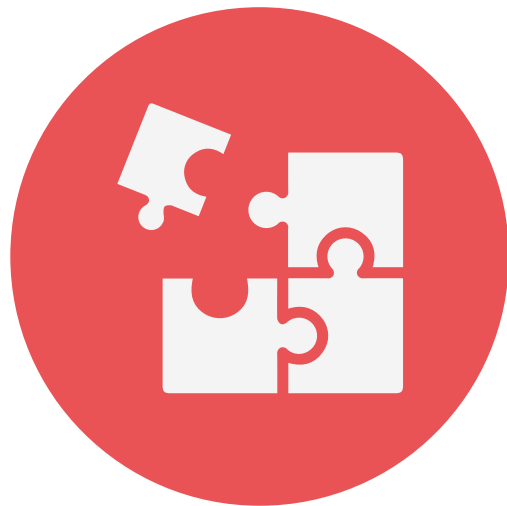
# 프로젝트 개요

기존 jsp 프로젝트를 MVC패턴을 적용한 JSP  
프로젝트의 형태로 변경한다.

# 팀 구성원 역할

---

## 팀 구성원의 역할



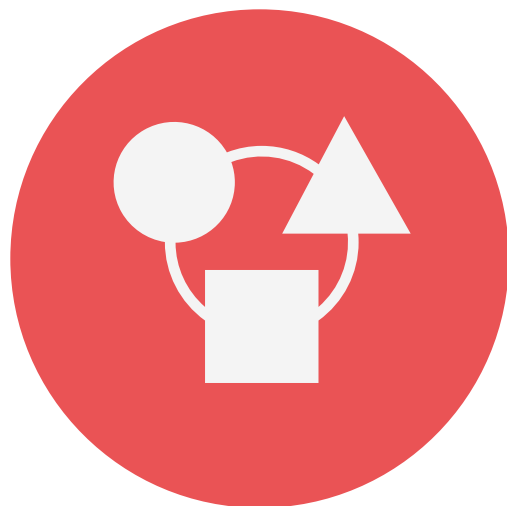
### 한형빈

게시판 구현  
DB데이터 테이블 및 데이터  
요구사항 정의서 작성



### 장순재

검색 기능 구현  
콘텐츠 디자인 구성  
ppt작성



### 추형욱

카테고리 기능 구현  
데이터 수정, 삭제  
발표



### 이지환

세션 로그인,  
페이징 기능 추가  
Css 요소 추가  
JSP MVC 패턴 변경

# 기존 프로젝트

---

# DATABASE

User					
🔑	user_num	Domain	int	NOT NULL	auto_increment
	uesr_id	Domain	varchar(60)	NOT NULL	Comment
	passwd	Domain	varchar(100)	NOT NULL	Comment
	name	Domain	varchar(100)	NOT NULL	Comment
	nickname	Domain	varchar(100)	NOT NULL	Comment
	email	Domain	varchar(100)	NOT NULL	Comment
	phone	Domain	varchar(60)	NULL	Comment
	rank	Domain	int	NOT NULL	기본값 1 (관리자는 0)
	headLine	Domain	varchar(100)	NOT NULL	Comment

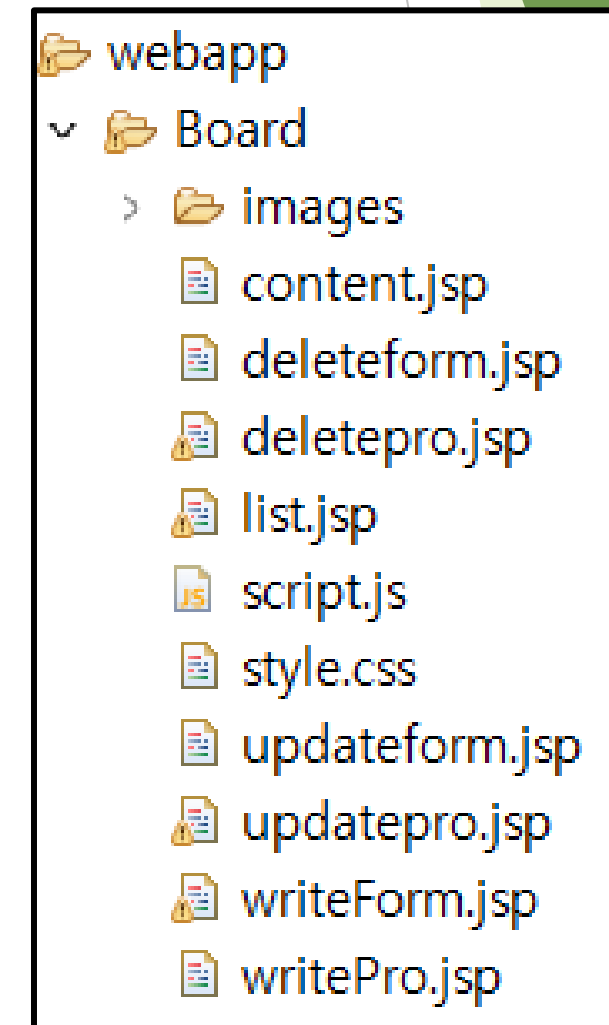
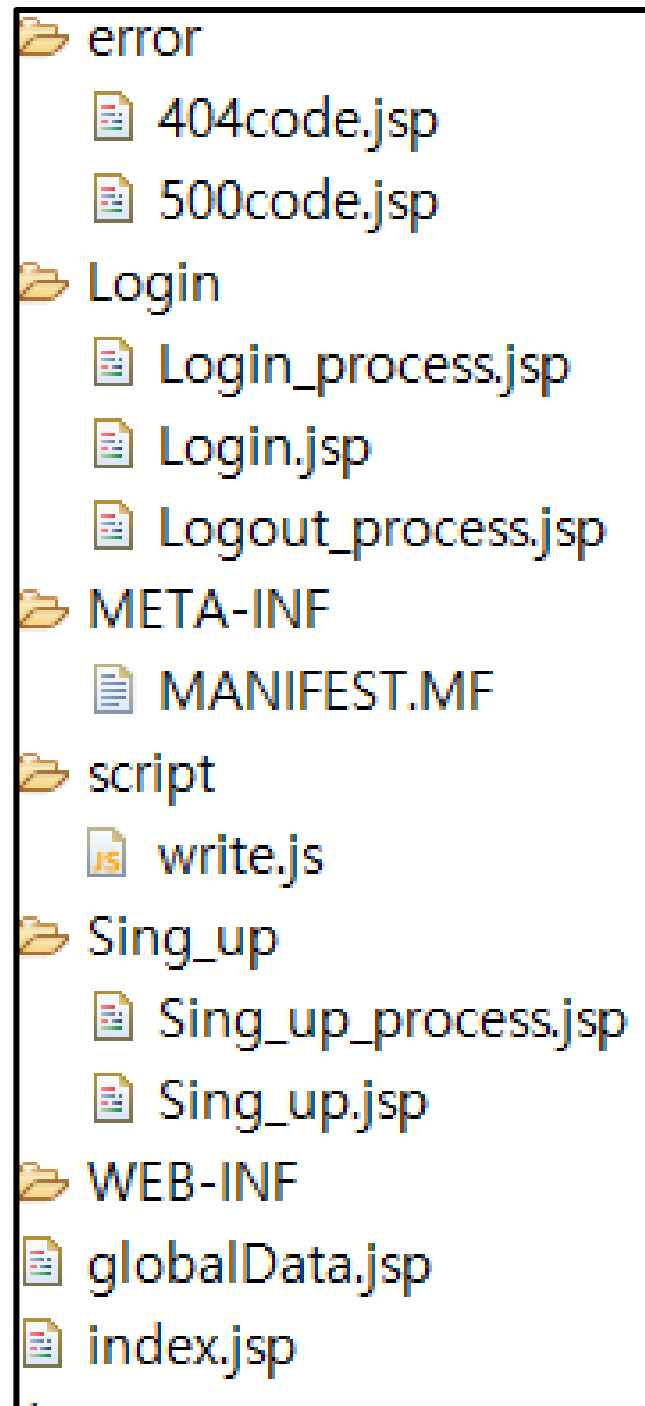
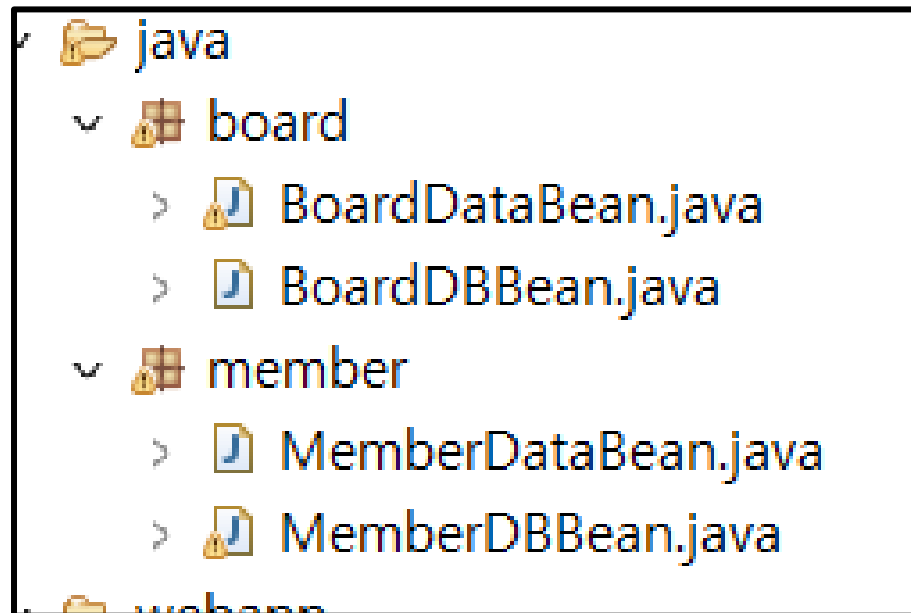


board					
🔑	board_num	Domain	int	NOT NULL	auto_incerment
🔑	writer	Domain	int	NOT NULL	닉네임을 가져옴
	subject	Domain	varchar(50)	NOT NULL	Comment
	reg_date	Domain	datetime	NOT NULL	기본값 : 현재 시간
	readcount	Domain	int	NULL	기본값:0
	ref	Domain	int	NULL	Comment
	re_step	Domain	smallint(6)	NULL	Comment
	re_level	Domain	smallint(6)	NULL	Comment
	content	Domain	text	NULL	Comment
	ip	Domain	varchar(20)	NULL	Comment



# 파일구조

기능별로 폴더를 따로 나누어서 가독성 및 모듈화



# INDEX.JSP

## 시작 하는 부분

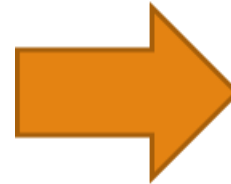
세션기능으로 로그인 한후에는 실행시 게시판으로 바로 넘어감

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
2 <%@ include file="../globalData.jsp" %>
3 <%
4     // 세션에서 로그인 상태를 확인합니다.
5     String loginState = (String) session.getAttribute("loginState");
6
7     // 로그인 상태가 없거나, 로그인 상태가 없는 경우 로그인 페이지로 이동합니다.
8     if (loginState == null || loginState.isEmpty()) {
9         response.sendRedirect("../Login/Login.jsp");
10        return;
11    }
12
13    // 로그인 상태가 있으면 상태에 따라 적절한 페이지로 리다이렉트합니다.
14    if (loginState.equals("admin")) {
15        response.sendRedirect("../Board/list.jsp");
16    } else if (loginState.equals("common")) {
17        response.sendRedirect("../Board/list.jsp");
18    } else {
19        // 그 외의 상황에 대한 처리(예를 들어, 세션이 손상된 경우)
20        response.sendRedirect("../Login/Login.jsp");
21    }
22 %>
```

# GLOBALDATA.JSP

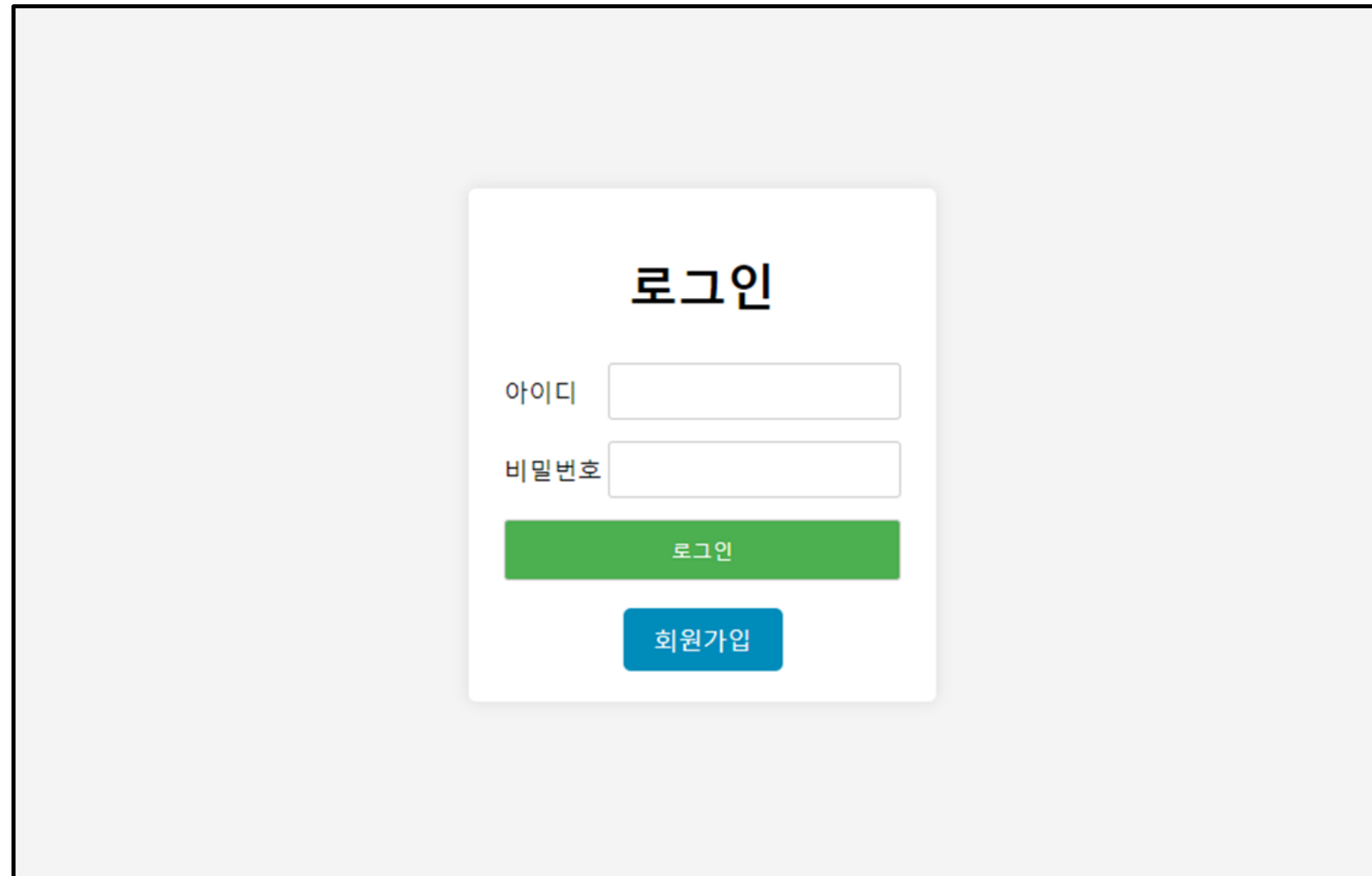
파일의 경로를 지정하는 부분  
다른 폴더에 있는 파일의 경로를 넘어가게 함

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"  
2   pageEncoding="UTF-8"%>  
3 <%@ page import="javax.servlet.http.HttpSession" %>  
4 <%@ page import="member.MemberDataBean" %>  
5 <%!  
6   static final String RootDir="/Used_trade_community";  
7   static final String Loginroot = RootDir + "/Login";  
8   static final String Singuproot = RootDir + "/Sing_up";  
9   static final String boardroot = RootDir + "/Board";  
10  
11  
12 %>
```



```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"  
2   pageEncoding="UTF-8"%>  
3 <%@ include file="../globalData.jsp"%>  
4 <!DOCTYPE html>  
5 <html>  
6 <head>  
7   <meta charset="UTF-8">  
8   <title>Register</title>  
9   <link href="../css/Sing_up.css" rel="stylesheet" type="text/css">  
10 </head>  
11 <body>  
12   <div class="register-container">  
13     <h1>회원가입</h1>  
14     <form action="<%=Singuproot%>/Sing_up_process.jsp" method="post">  
15       <table>  
16         <tr>  
17           <td>이름</td>  
18           <td><input type="text" name="name"></td>  
19         </tr>
```

# 로그인 폼



The image shows a login form centered on a light gray background. The form is a white rectangle with rounded corners. At the top of the form, the word '로그인' (Login) is written in black. Below it, there are two input fields. The first is labeled '아이디' (ID) and the second is labeled '비밀번호' (Password). Both labels are in black text to the left of their respective input boxes. Below the password field is a green button with the text '로그인' (Login) in white. Below the green button is a blue button with the text '회원가입' (Sign Up) in white. The entire form is enclosed in a thin black border.

세션을 통한 로그인,  
아이디, 비밀번호 동일 여부 자바스크립트 구현



# LOGIN\_PROCESS.JSP

<%

```
request.setCharacterEncoding("UTF-8");
```

```
String id = request.getParameter("id");
```

```
String passwd = request.getParameter("passwd");
```

```
MemberDBBean db = MemberDBBean.getInstance();
```

```
int loginResult = db.login(id, passwd);
```

```
MemberDataBean member = db.getMemberInfo(id); // 회원 정보 가져오기
```

```
if (loginResult == 2) {
```

```
    // 일반 로그인 성공 시 세션에 로그인 상태를 저장
```

```
    session.setAttribute("loginState", "common");
```

```
    session.setAttribute("userId", member.getId());
```

```
    session.setAttribute("userNickname", member.getNickname());
```

```
    session.setAttribute("userPassword", member.getPasswd());
```

```
    session.setAttribute("userMemNum", member.getMem_num());
```

```
    session.setAttribute("userEmail", member.getEmail()); // 회원 번호 추가
```

```
    response.sendRedirect("../Board/list.jsp");
```



```

} else if (loginResult == 1) {
    // 관리자 로그인 성공 시 세션에 관리자 로그인 상태를 저장
    session.setAttribute("loginState", "admin");
    session.setAttribute("userId", member.getId());
    session.setAttribute("userNickname", member.getNickname());
    session.setAttribute("userPassword", member.getPasswd());
    session.setAttribute("userMemNum", member.getMem_num());
    session.setAttribute("userEmail", member.getEmail()); //회원 번호 추가
    response.sendRedirect("../Board/list.jsp"); // 관리자 페이지로 리다이렉트
} else if (loginResult == 3) {
    // 아이디가 존재하지 않는 경우
    %>
    <script>
        alert("존재하지 않는 아이디입니다.");
        window.location.href = "../Login.jsp";
    </script>
    <%
} else if (loginResult == 4) {
    // 비밀번호가 틀린 경우
    %>
    <script>
        alert("비밀번호가 잘못되었습니다.");
        window.location.href = "../Login.jsp";
    </script>
    <%
} else {
    // 기타 로그인 실패 시 세션 제거 및 로그인 실패 페이지로 리다이렉트
    session.invalidate(); // 세션 제거
    response.sendRedirect("../Login_fail.jsp");
}
%>

```

# 회원가입 폼

## 회원가입

이름

아이디

비밀번호

닉네임

이메일

전화번호

주민등록번호

가입

이미 회원이신가요? [로그인 페이지로 이동](#)

```
1 function writeSave(){
2     var writeform = document.writeform;
3
4     if(!writeform.writer.value){
5         alert("작성자를 입력하십시오.");
6         writeform.writer.focus();
7         return false;
8     }
9
10    if(!writeform.headline.value){
11        alert("카테고리를 입력하세요");
12        writeform.headline.focus();
13        return false;
14    }
15
16    if(!writeform.subject.value){
17        alert("제목을 입력하십시오.");
18        writeform.subject.focus();
19        return false;
20    }
21
22    if(!writeform.content.value){
23        alert("내용을 입력하십시오.");
24        writeform.content.focus();
25        return false;
26    }
27
28    if(!writeform.passwd.value){
29        alert("비밀번호를 입력하십시오.");
30        writeform.passwd.focus();
31        return false;
32    }
33 };
```

## 자바 스크립트를 통해 null 입력 방지



# SING\_UP\_PROCESS.JSP

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ include file = "../globalData.jsp" %>
<%@ page import="member.MemberDataBean, member.MemberDBBean" %>
<%
    request.setCharacterEncoding("UTF-8");

    MemberDataBean member = new MemberDataBean();
    member.setName(request.getParameter("name"));
    member.setId(request.getParameter("id"));
    member.setPasswd(request.getParameter("passwd"));
    member.setNickname(request.getParameter("nickname"));
    member.setEmail(request.getParameter("email"));
    member.setPhone(request.getParameter("phone"));
    member.setRRN(request.getParameter("RRN"));
    member.setRanks(1); // 기본값 설정

    MemberDBBean db = MemberDBBean.getInstance();
    db.insertMember(member);

%>

<script>
    alert("회원가입 되었습니다.");
    window.location.href = "../Login/Login.jsp";
</script>
```



# 게시판 생성 , 페이징 기능 추가

번호	제 목	작성자	작성일
1	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
2	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
3	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
4	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
5	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
6	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
7	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
8	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
9	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24
10	[사고팝니다.] 첫 번째 게시물입니다.	관리자	2024-04-19 05:24

[1][2][3][4][5][6][7][8][9][10][다음]

검색 유형 선택 ▼    검색어 입력    

검색하기

    초기화

# 검색기능 구현

제목, 작성자, 내용을 이용한 검색기능구현

검색 기능 추가 전

```
public List<BoardDataBean> getArticles(int start, int end) throws Exception {  
    Connection conn = null;  
    PreparedStatement pstmt = null;  
    ResultSet rs = null;  
    List<BoardDataBean> articleList = null;  
    try {  
        conn = getConnection();  
  
        pstmt = conn.prepareStatement("select * from board order by ref desc, re_step asc limit ?,? ");  
        pstmt.setInt(1, start - 1);  
        pstmt.setInt(2, end);  
        rs = pstmt.executeQuery();  
    }  
}
```



# 검색 기능 추가 후

```
public List<BoardDataBean> getArticles(int start, int end, String type, String keyword) throws Exception {
    Connection conn = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    List<BoardDataBean> articleList = null;
    try {
        conn = getConnection();
        if (keyword == "") {
            pstmt = conn.prepareStatement("select * from board order by ref desc, re_step asc limit ?,? ");
            pstmt.setInt(1, start - 1);
            pstmt.setInt(2, end);
        } else {
            if (type.equals("subject")) {
                pstmt = conn.prepareStatement(
                    "select * from board where subject like ? order by ref desc, re_step asc limit ?,? ");
            } else if (type.equals("writer")) {
                pstmt = conn.prepareStatement(
                    "select * from board where writer like ? order by ref desc, re_step asc limit ?,? ");
            } else if (type.equals("content")) {
                pstmt = conn.prepareStatement(
                    "select * from board where content like ? order by ref desc, re_step asc limit ?,? ");
            }
            // pstmt.setString(1, type);
            pstmt.setString(1, "%" + keyword + "%");
            pstmt.setInt(2, start - 1);
            pstmt.setInt(3, end);
        }
    }
}
```

# 조회수와 답변기능, 수정, 삭제 권한 **구현**

번호	제목	작성자	작성일	조회	IP
1	[사고팝니다] dd 	admin	2024-05-20 05:17	20	0:0:0:0:0:0:0:1
2	[사고팝니다]  [답변]	admin	2024-05-20 05:17	1	0:0:0:0:0:0:0:1
3	[핫딜특가]  [답변] fff	admin	2024-05-20 05:17	1	0:0:0:0:0:0:0:1

글내용 보기

117	조회수	21
admin	작성일	2024-05-20 05:17
dd		
<div><div>글수정</div><div>글삭제</div><div>답글쓰기</div><div>글목록</div></div>		

```
<%  
if ("common".equals(loginState) && userNickname.equals(article.getWriter())) || "admin".equals(log  
%>  
  
    <input type="button" value="글수정" onclick="document.location.href='modifyPage.bo?num=  
        &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&";  
    <input type="button" value="글삭제" onclick="document.location.href='deletePage.bo?num=  
        &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~";  
    <%  
}  
%>  
  
    <input type="button" value="답글쓰기" onclick="document.location.href='writePage.bo?num:  
        &nbsp;&nbsp;&~";  
    <input type="button" value="글목록" onclick="document.location.href='list.bo?pageNum=<~"  
    </td>  
    </tr>  
</table>  
<%
```

조회수가 특정값을 넘을때 [ $\geq 20$ ] Hot이라는 이미지가 뜨게 되고  
특정 글에 띄어쓰기와 이미지를 통한 답글 구현  
로그인형태에 따라[관리자, 작성자] 수정 삭제 구현

# MVC 프로젝트 변경 수행 절차

---

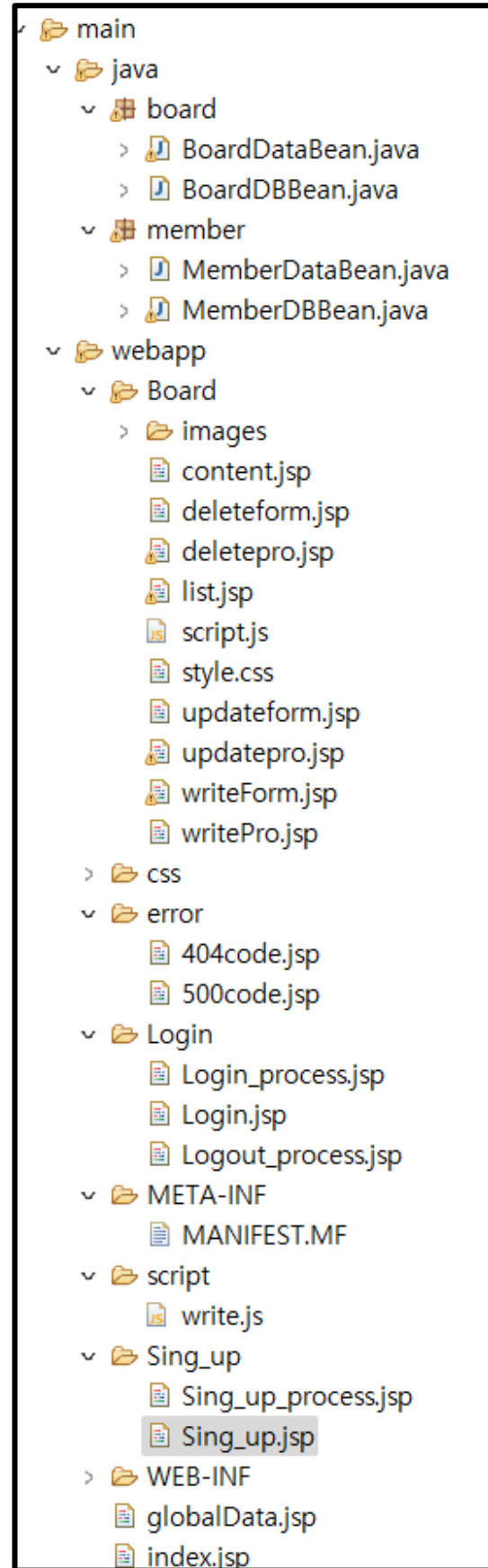
# MVC 프로젝트 변경 수행 절차

- ※ 구조 변경 [DAO, DTO 형태로 변경]
- ※ 테이블에 따라 컨트롤러 생성
- ※ Pro.jsp => command 형태로 변경
- ※ command 조정 (javascript 적용 오류 수정)
- ※ View 페이지 경로 변경 및 실행 전체적 변경

# MVC 프로젝트로의 변경점

---

# 프로젝트 구조 변경



**DataBean => DTO**

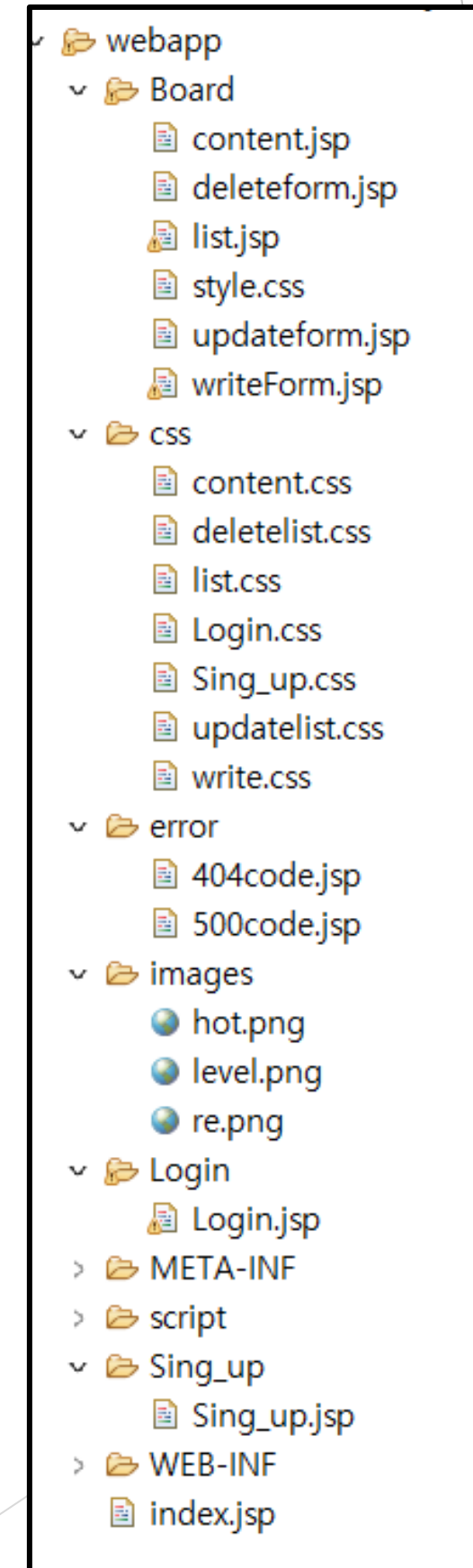
**DBBean => DAO**

**Controller 추가**

**Pro => Command**

**서블릿이 생겼으므로  
GlobalData.jsp는 삭제**

**Before**



**After**



# DB 테이블마다 컨트롤러를 나눔

## Member 테이블 서블릿 컨트롤러(\*.mo)

```
22 @WebServlet("*.mo")
23 public class memberController extends HttpServlet {
24     private static final long serialVersionUID = 1L;
25
26     /**
27      * @see HttpServlet#HttpServlet()
28      */
29     public memberController() {
30         super();
31         // TODO Auto-generated constructor stub
32     }
33
34     /**
35      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
36      */
37     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException {
38         // TODO Auto-generated method stub
39         //response.getWriter().append("Served at: ").append(request.getContextPath());
40         doAction(request, response);
41     }
42
43     /**
44      * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
45      */
46     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException {
47         // TODO Auto-generated method stub
48         doGet(request, response);
49     }
50 }
```

## Board 테이블 서블릿 컨트롤러(\*.bo)

```
24 @WebServlet("*.bo")
25 public class boardController extends HttpServlet {
26     private static final long serialVersionUID = 1L;
27
28     /**
29      * @see HttpServlet#HttpServlet()
30      */
31     public boardController() {
32         super();
33         // TODO Auto-generated constructor stub
34     }
35
36     /**
37      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
38      */
39     protected void doGet(HttpServletRequest request, HttpServletResponse response) throws ServletException {
40         // TODO Auto-generated method stub
41         //response.getWriter().append("Served at: ").append(request.getContextPath());
42         doAction(request, response);
43     }
44
45     /**
46      * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse response)
47      */
48     protected void doPost(HttpServletRequest request, HttpServletResponse response) throws ServletException {
49         // TODO Auto-generated method stub
50         doGet(request, response);
51     }
52 }
```

# 컨트롤러 요소 추가[command, viewpage]

```
1 package member.command;
2
3 import javax.servlet.http.HttpServletRequest;
4
5
6 public interface memberCommand {
7
8     void execute(HttpServletRequest request, HttpServletResponse response);
9
10 }
11
```

```
1 package board.command;
2
3 import javax.servlet.http.HttpServletRequest;
4
5
6 public interface boardCommand {
7
8     void execute(HttpServletRequest request, HttpServletResponse response);
9
10 }
11
```

```
protected void doAction(HttpServletRequest request, HttpServletResponse response) throws ServletException {
    // TODO Auto-generated method stub
    request.setCharacterEncoding("utf-8");
    String viewPage = null; //어떤 페이지로 갈 건지 지정
    memberCommand command = null; //다형성 이용, view 값에 따라서 목록 조회 등 수행 예정
    //ctrl shift o = 자동 임포트 및 안 쓰는 임포트 삭제 기능

    //주소 가져 오는 것
    String uri = request.getRequestURI();
    String conPath = request.getContextPath();
    String com = uri.substring(conPath.length());
    //com : /list.do, /modify.do 등... 커맨드만 남음

    //ctrl shift o = 자동 import 및 불필요한 import 삭제
    //ctrl shift s 누르고 r = getter setter 자동 생성
    //ctrl t = f3이랑 비슷하지만 어떤 선언부로 이동할 지 선택이 가능함

    //Link With Editor -> 현재 작업하고 있는 걸 자동 선택해 줌
    //점 세개 -> Package Presentation
    //ctrl shift r = 해당 위치로 이동
}
```

```
protected void doAction(HttpServletRequest request, HttpServletResponse response) throws ServletException {
    // TODO Auto-generated method stub
    request.setCharacterEncoding("utf-8");
    String viewPage = null; //어떤 페이지로 갈 건지 지정
    boardCommand command = null; //다형성 이용, view 값에 따라서 목록 조회 등 수행 예정
    //ctrl shift o = 자동 임포트 및 안 쓰는 임포트 삭제 기능

    //주소 가져 오는 것
    String uri = request.getRequestURI();
    String conPath = request.getContextPath();
    String com = uri.substring(conPath.length());
    //com : /list.do, /modify.do 등... 커맨드만 남음

    //ctrl shift o = 자동 import 및 불필요한 import 삭제
    //ctrl shift s 누르고 r = getter setter 자동 생성
    //ctrl t = f3이랑 비슷하지만 어떤 선언부로 이동할 지 선택이 가능함

    //Link With Editor -> 현재 작업하고 있는 걸 자동 선택해 줌
    //점 세개 -> Package Presentation
    //ctrl shift r = 해당 위치로 이동
}
```

command = command의 따라 사용할 커맨드 클래스를 지정

ViewPage = 이동할 페이지

Com = view(jsp) 에서 적은 페이지를 가져오고(uri) 그것을 서블릿 경로를(conpath)읽어 들여 가져온 값

```
@WebServlet("*.mo")
```

```
@WebServlet("*.bo")
```

# 컨트롤러 요소 추가[command, viewpage]

## 로그인 관련 이동 경로 및 command

```
if(com.equals("/LoginPage.mo")) {
    viewPage = "/Login/Login.jsp"; // 로그인 페이지로 이동
} else if(com.equals("/Login.mo")) {
    command = new LoginCommand();
    command.execute(request, response);
    return; // 로그인 후 리다이렉트는 LoginCommand에서 처리
} else if(com.equals("/Logout.mo")) {
    viewPage = "/LoginPage.mo"; // 로그아웃 후 로그인 페이지로 이동
    command = new LogoutCommand();
    command.execute(request, response);
} else if(com.equals("/SingupPage.mo")) {
    viewPage = "/Sing_up/Sing_up.jsp"; // 회원가입 페이지로 이동
} else if(com.equals("/Singup.mo")) {
    command = new SingUpCommand();
    command.execute(request, response);
    return; // SingUpCommand에서 리다이렉트를 처리
}

if(viewPage.equals("/list.bo")) {
    response.sendRedirect(request.getContextPath() + viewPage); // 페이지 리다이렉트
} else {
    RequestDispatcher dispatcher = request.getRequestDispatcher(viewPage);
    dispatcher.forward(request, response); // 페이지 포워딩
}
```

View(Jsp) 에서 적은 이동경로에 따라  
경로에 따라 서블릿에 연결 후  
페이지를 이동하거나 Command를  
실행 후 경로를 지정해 준다.  
한 Command를 실행 중에 조건에 따라 다른 경로로 이동하는 경우에는  
[ex ] adminpage 등  
viewpage를 없애고 command에서 이동

## 게시판 관련 이동 경로 및 comand

```
if(com.equals("/list.bo")) {
    viewPage = "/Board/list.jsp";
    command = new boardListCommand();
    command.execute(request, response);
} else if(com.equals("/content.bo")) {
    viewPage = "/Board/content.jsp";
    command = new boardContentCommand();
    command.execute(request, response);

} else if(com.equals("/modifyPage.bo")) {
    viewPage = "/Board/updateform.jsp";
} else if(com.equals("/modify.bo")) {
    viewPage = "/list.bo";
    command = new boardUpdateCommand();
    command.execute(request, response);

} else if(com.equals("/writePage.bo")) {
    viewPage = "/Board/writeForm.jsp";
} else if(com.equals("/write.bo")) {
    viewPage = "/list.bo";
    command = new boardInputCommand();
    command.execute(request, response);

} else if(com.equals("/deletePage.bo")) {
    viewPage = "/Board/deleteform.jsp"; //답글 화면
} else if(com.equals("/delete.bo")) {
    viewPage = "/list.bo";
    command = new boardDeleteCommand();
    command.execute(request, response);
}

if(viewPage.equals("/list.bo")) {
    response.sendRedirect(conPath+viewPage);
} else {
    RequestDispatcher dispatcher =
        request.getRequestDispatcher(viewPage);
    dispatcher.forward(request, response);
}
```



# 프로세스 실행 [서비스] 형태 변경

기존 jsp에서 실행했던 프로세스의 실행을 jsp에서 java 클래스에서 실행

## loginpro.jsp

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8"
2   pageEncoding="UTF-8"%>
3 <%@ include file = "../globalData.jsp" %>
4 <%@ page import="member.MemberDBBean, member.MemberDataBean" %>
5 <%
6   request.setCharacterEncoding("UTF-8");
7
8   String id = request.getParameter("id");
9   String passwd = request.getParameter("passwd");
10
11   MemberDBBean db = MemberDBBean.getInstance();
12   int loginResult = db.login(id, passwd);
13   MemberDataBean member = db.getMemberInfo(id); // 회원 정보 가져오기
14
15   if (loginResult == 2) {
16     // 일반 로그인 성공 시 세션에 로그인 상태를 저장
17     session.setAttribute("loginState", "common");
18     session.setAttribute("userId", member.getId());
19     session.setAttribute("userNickname", member.getNickname());
20     session.setAttribute("userPassword", member.getPasswd());
21     session.setAttribute("userMemNum", member.getMem_num());
22     session.setAttribute("userEmail", member.getEmail()); // 회원 번호 추가
23     response.sendRedirect("../Board/list.jsp");
24   } else if (loginResult == 1) {
25     // 관리자 로그인 성공 시 세션에 관리자 로그인 상태를 저장
26     session.setAttribute("loginState", "admin");
27     session.setAttribute("userId", member.getId());
28     session.setAttribute("userNickname", member.getNickname());
29     session.setAttribute("userPassword", member.getPasswd());
30     session.setAttribute("userMemNum", member.getMem_num());
31     session.setAttribute("userEmail", member.getEmail()); // 회원 번호 추가
32     response.sendRedirect("../Board/list.jsp"); // 관리자 페이지로 리다이렉트
```

```
33   } else if (loginResult == 3) {
34     // 아이디가 존재하지 않는 경우
35     %>
36     <script>
37       alert("존재하지 않는 아이디입니다.");
38       window.location.href = "../Login.jsp";
39     </script>
40     <%
41   } else if (loginResult == 4) {
42     // 비밀번호가 틀린 경우
43     %>
44     <script>
45       alert("비밀번호가 잘못되었습니다.");
46       window.location.href = "../Login.jsp";
47     </script>
48     <%
49   } else {
50     // 기타 로그인 실패 시 세션 제거 및 로그인 실패 페이지로 리다이렉트
51     session.invalidate(); // 세션 제거
52     response.sendRedirect("../Login_fail.jsp");
53   }
54 %>
```

Pro jsp를  
Java 클래스의 형태  
Command 로 변경

코드는 pro와 유사  
자바 스크립트 관련은  
out.println로 실행

# 프로세스 실행 [서비스] 형태 변경

# logincommand.java

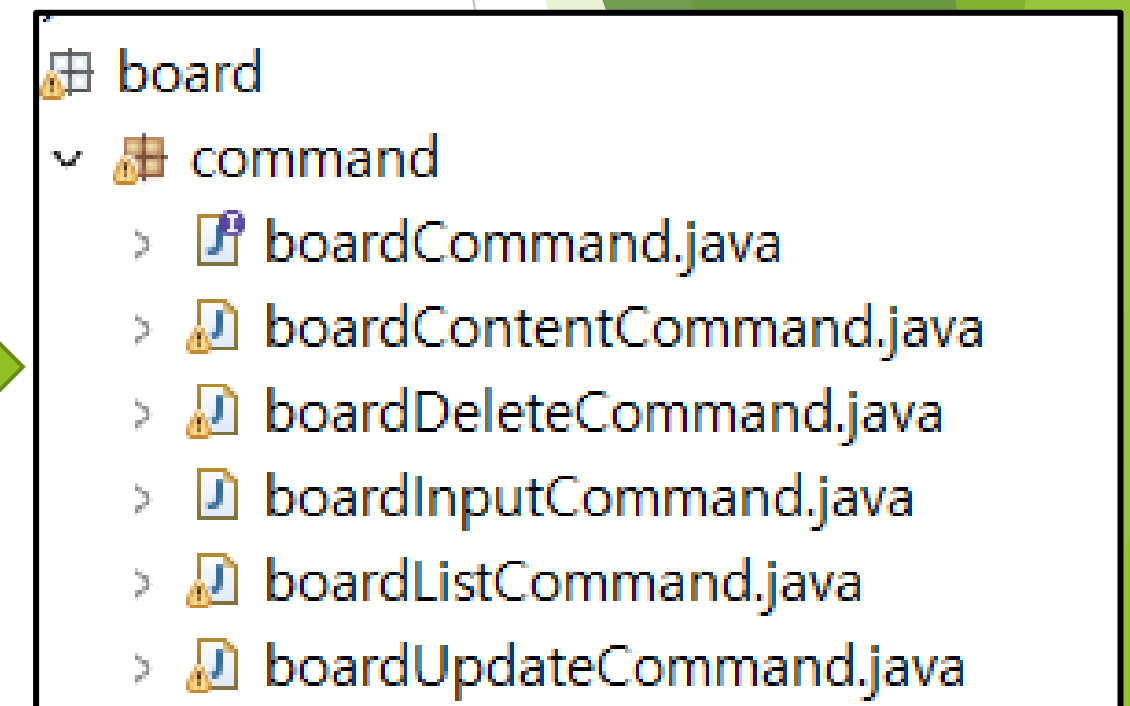
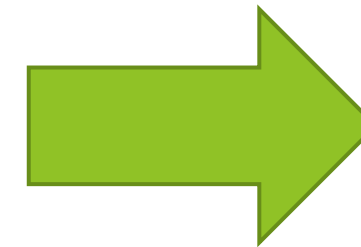
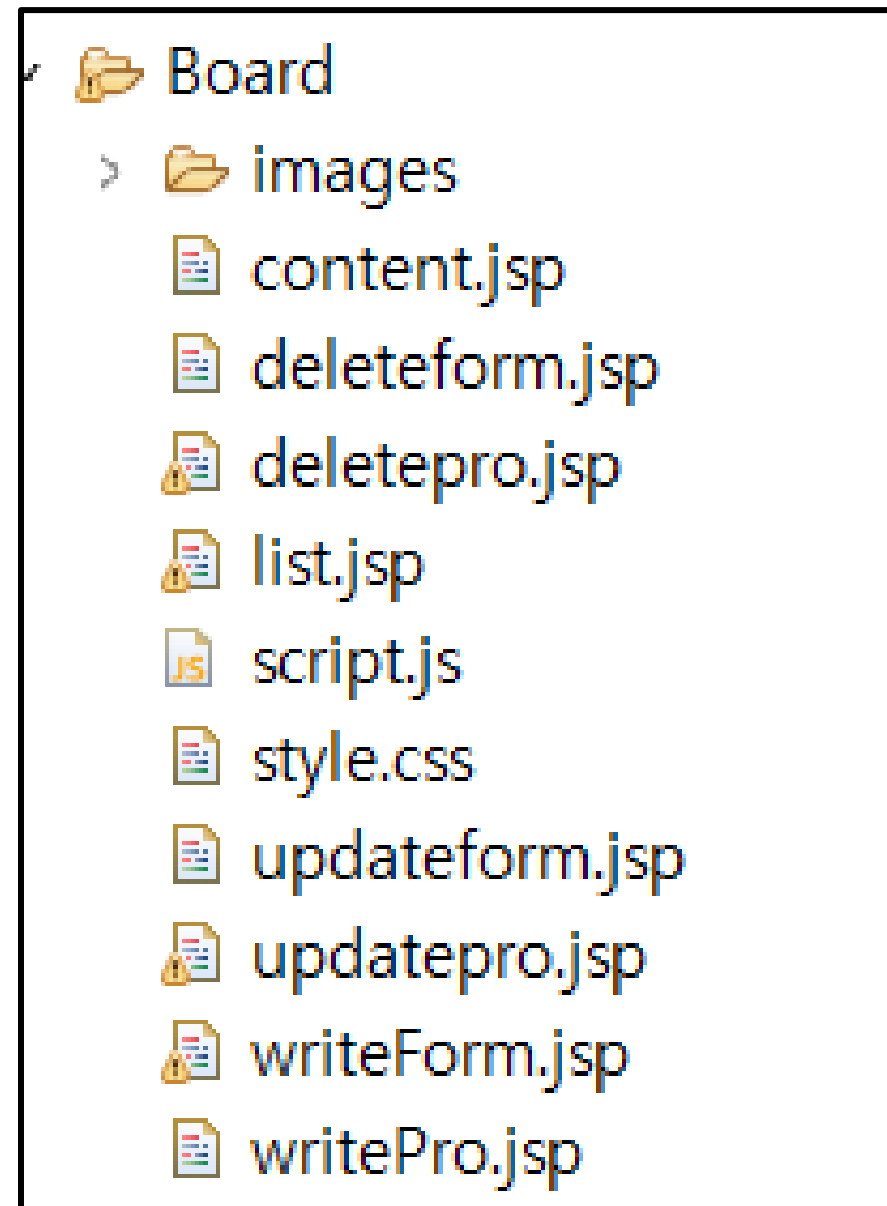
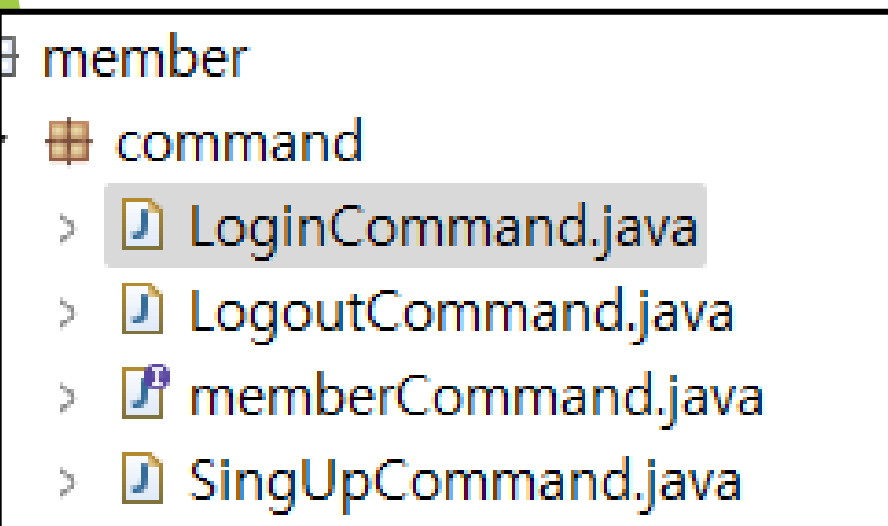
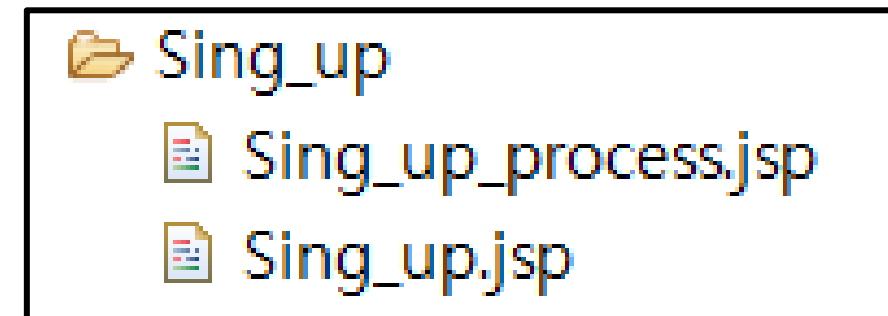
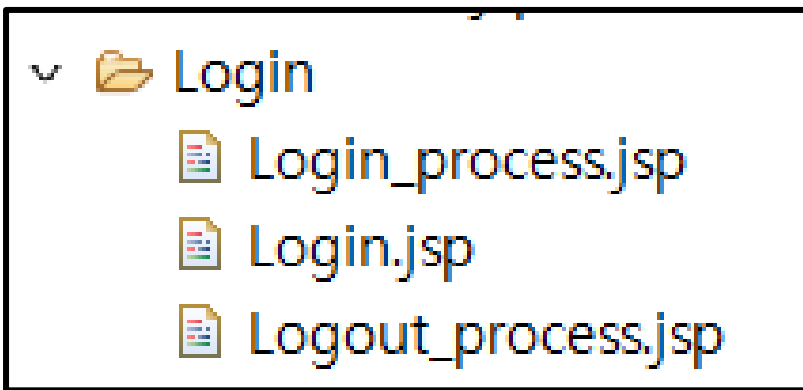
```
1 package member.command;
2
3 import java.io.PrintWriter;
4
5
6
7
8
9
10 public class LoginCommand implements memberCommand {
11
12     @Override
13     public void execute(HttpServletRequest request, HttpServletResponse response) {
14         try {
15             request.setCharacterEncoding("UTF-8");
16
17             String id = request.getParameter("id");
18             String passwd = request.getParameter("passwd");
19
20             memberDAO db = memberDAO.getInstance();
21             int loginResult = db.login(id, passwd);
22             memberDTO member = db.getMemberInfo(id);
23
24             HttpSession session = request.getSession();
25
26             if (loginResult == 2) {
27                 // Common user login successful
28                 session.setAttribute("loginState", "common");
29                 session.setAttribute("userId", member.getId());
30                 session.setAttribute("userNickname", member.getNickname());
31                 session.setAttribute("userPassword", member.getPasswd());
32                 session.setAttribute("userMemNum", member.getMem_num());
33                 session.setAttribute("userEmail", member.getEmail());
34                 response.sendRedirect("list.bo");
35             }
36         } catch (Exception e) {
37             // TODO Auto-generated catch block
38             e.printStackTrace();
39         }
40     }
41 }
```

```

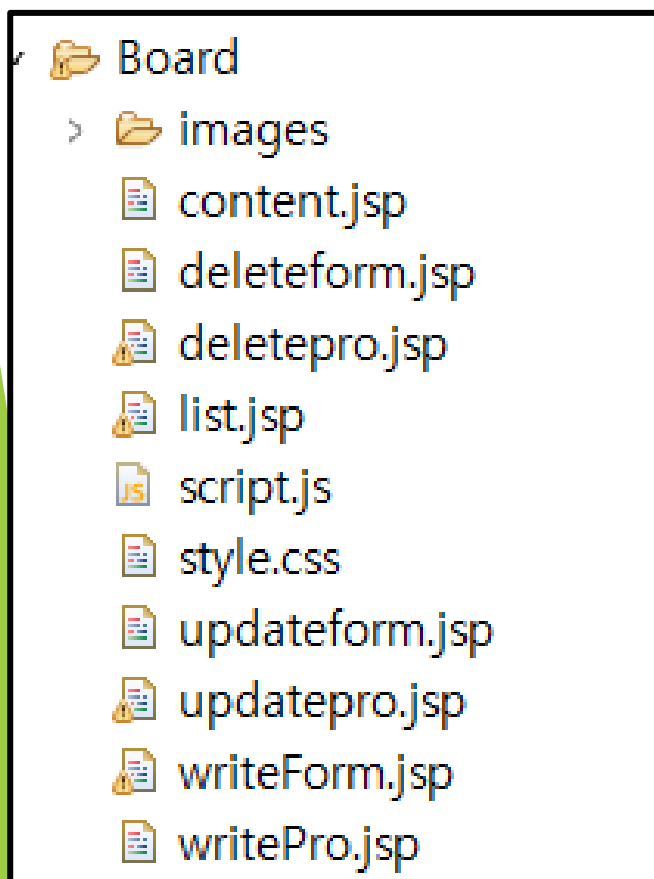
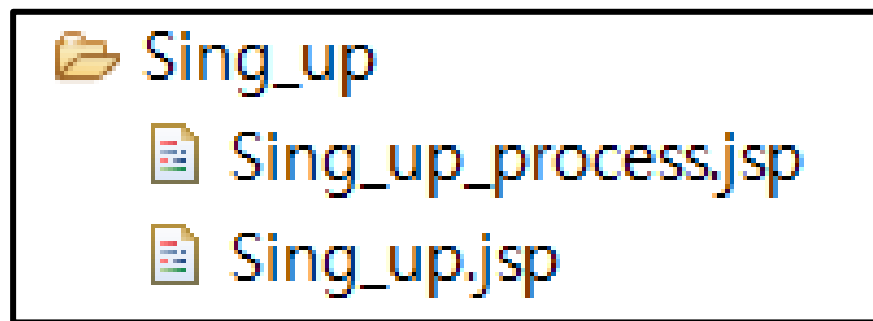
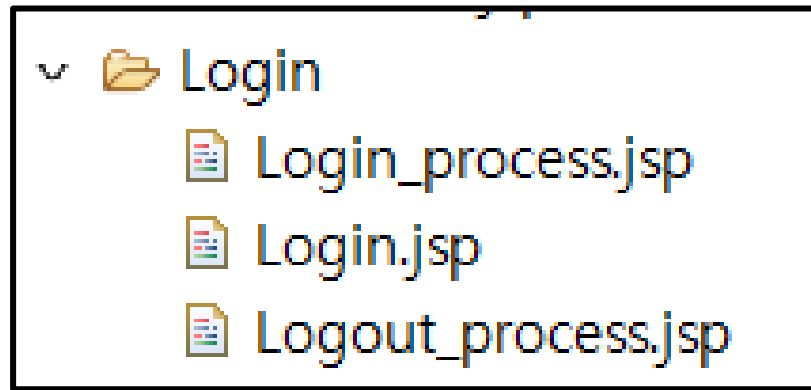
35     } else if (loginResult == 1) {
36         // Admin login successful
37         session.setAttribute("loginState", "admin");
38         session.setAttribute("userId", member.getId());
39         session.setAttribute("userNickname", member.getNickname());
40         session.setAttribute("userPassword", member.getPasswd());
41         session.setAttribute("userMemNum", member.getMem_num());
42         session.setAttribute("userEmail", member.getEmail());
43         response.sendRedirect("list.bo");
44     } else {
45         // Handle login failure cases
46         response.setContentType("text/html;charset=UTF-8");
47         PrintWriter out = response.getWriter();
48         out.println("<script>");
49         if (loginResult == 3) {
50             // ID does not exist
51             out.println("alert('존재하지 않는 아이디입니다.');

```

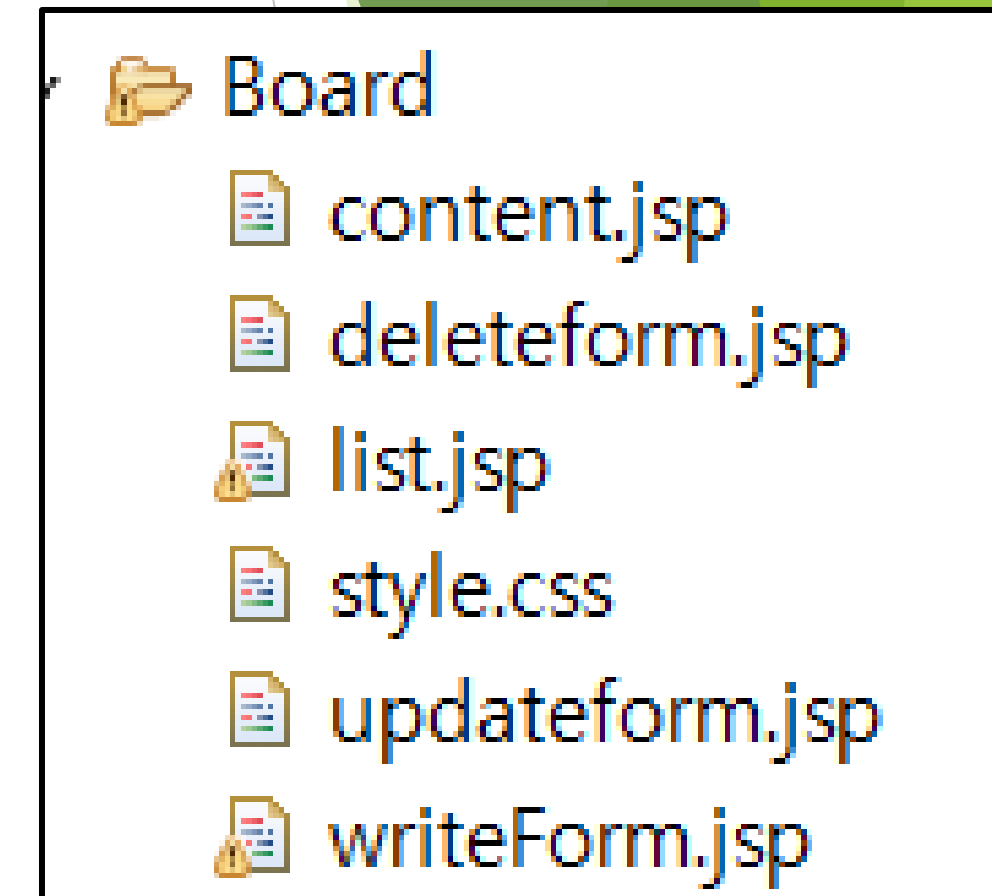
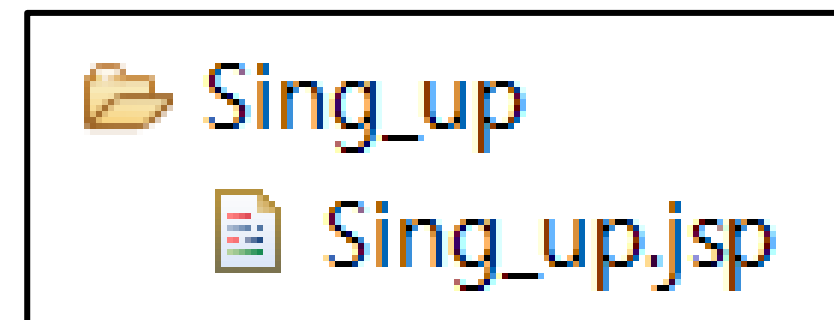
# 기존 pro를 command형식으로 변경



# 불필요한 pro부분을 제거



결과 적으로 form의 형태  
Jsp(View)만 남게 됨



# 기존 JSP(View)에 주소값을 변경

기존 경로를 서블릿 컨트롤러로 연결함

MEMBER 컨트롤러로 이동

```
1  <%@ page language="java" contentType="text/html" %>
2  pageEncoding="UTF-8" isELIgnored="false"%>
3
4  <!DOCTYPE html>
5  <html>
6  <head>
7      <meta charset="UTF-8">
8      <title>Login</title>
9      <link href="${pageContext.request.contextPath}/resources/css/login.css" rel="stylesheet">
10
11 </head>
12 <body>
13     <div class="login-container">
14         <h1>로그인</h1>
15         <form action="Login.mo" method="post">
16             <table>
17                 <tr>
18                     <td>아이디</td>
19                     <td><input type="text" name="id"></td>
20                 </tr>
21                 <tr>
22                     <td>비밀번호</td>
23                     <td><input type="password" name="password"></td>
24                 </tr>
25                 <tr>
26                     <td colspan="2"><input type="button" value="로그인"></td>
27                 </tr>
28             </table>
29         </form>
30     </div>
31 </body>
32 </html>
```

BOARD 컨트롤러로 이동

```
if (startPage > 10) { %>
<a href="list.bo?pageNum=<%= startPage - 10 %>&type=<%=field%>&keyword=<%=query%>">[이전]
<%
}

for (int i = startPage ; i <= endPage ; i++) { %>
<a href="list.bo?pageNum=<%= i %>&type=<%=field%>&keyword=<%=query%>">[<%= i %>]
</a>
<%
}

if (endPage < pageCount) { %>
<a href="list.bo?pageNum=<%= startPage + 10 %>&type=<%=field%>&keyword=<%=query%>">[다음]
<%
}
```



# JSP에 view 일부 내용 변경

DAO 부분에 GetAricle(num) 부분은 조회수를 추가해주는 코드이다.

Command쪽에서 이미 getatice(num)을 추가했기 때문에

기존 command.jsp에서 실행될때마다 getatice(num)이 발생하므로 이부분을 지우지 않으면 중복이 되어 조회수가 +2가 되게 된다.

```
// 글의 내용을 보기(1개의 글)(select문)<=content.jsp페이지에서 사용
public boardDTO getArticle(int num) throws Exception {
    Connection conn = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;
    boardDTO article = null;
    try {
        conn = getConnection();

        pstmt = conn.prepareStatement("update board set readcount=readcount+1 where num = ?");
        pstmt.setInt(1, num);
        pstmt.executeUpdate();

        pstmt = conn.prepareStatement("select * from board where num = ?");
        pstmt.setInt(1, num);
        rs = pstmt.executeQuery();

        if (rs.next()) {
            article = new boardDTO();
            article.setNum(rs.getInt("num"));
            article.setWriter(rs.getString("writer"));
            article.setEmail(rs.getString("email"));
            article.setSubject(rs.getString("subject"));
            article.setPasswd(rs.getString("passwd"));
            article.setReg_date(rs.getTimestamp("reg_date"));
            article.setReadcount(rs.getInt("readcount"));
            article.setRef(rs.getInt("ref"));
            article.setRe_step(rs.getInt("re_step"));
            article.setRe_level(rs.getInt("re_level"));
            article.setContent(rs.getString("content"));
            article.setIp(rs.getString("ip"));
            article.setHeadline(rs.getString("headline"));
        }
    } catch (Exception ex) {
        ex.printStackTrace();
    } finally {
        if (rs != null)
            rs.close();
        if (pstmt != null)
            pstmt.close();
        if (conn != null)
            conn.close();
    }
}
```

```
12 public class boardContentCommand implements boardCommand {
13     @Override
14     public void execute(HttpServletRequest request, HttpServletResponse response) {
15         try {
16             int num = Integer.parseInt(request.getParameter("num"));
17             String pageNum = request.getParameter("pageNum");
18
19             SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm");
20
21             boardDAO dbPro = boardDAO.getInstance();
22             boardDTO article = dbPro.getArticle(num);
23
24             int ref = article.getRef();
25             int re_step = article.getRe_step();
26             int re_level = article.getRe_level();
27
28             HttpSession session = request.getSession();
29             String userNickname = (String) session.getAttribute("userNickname");
30             String loginState = (String) session.getAttribute("loginState");
31
32             request.setAttribute("article", article);
33             request.setAttribute("userNickname", userNickname);
34             request.setAttribute("loginState", loginState);
35             request.setAttribute("pageNum", pageNum);
36         } catch (NumberFormatException e) {
37             e.printStackTrace();
38             // 게시글 번호 파싱 에러 처리
39         } catch (Exception e) {
40             e.printStackTrace();
41             // 기타 예외 처리
42         }
43     }
44 }
```

```
<%
int num = Integer.parseInt(request.getParameter("num"));
String pageNum = request.getParameter("pageNum");

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm");

try{
    BoardDBBean dbPro = BoardDBBean.getInstance();
    BoardDataBean article = dbPro.getArticle(num);

    int ref=article.getRef();
    int re_step=article.getRe_step();
    int re_level=article.getRe_level();

    String userNickname = (String) session.getAttribute("userNickname");
    String userEmail = (String) session.getAttribute("userEmail");
    String loginState = (String) session.getAttribute("loginState");
}%>
```

```
<%
int num = Integer.parseInt(request.getParameter("num"));
String pageNum = request.getParameter("pageNum");

SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd HH:mm");

try{
    boardDTO article = (boardDTO) request.getAttribute("article");

    int ref=article.getRef();
    int re_step=article.getRe_step();
    int re_level=article.getRe_level();

    String userNickname = (String) session.getAttribute("userNickname");
    String userEmail = (String) session.getAttribute("userEmail");
    String loginState = (String) session.getAttribute("loginState");
}%>
```

**THANK YOU.**