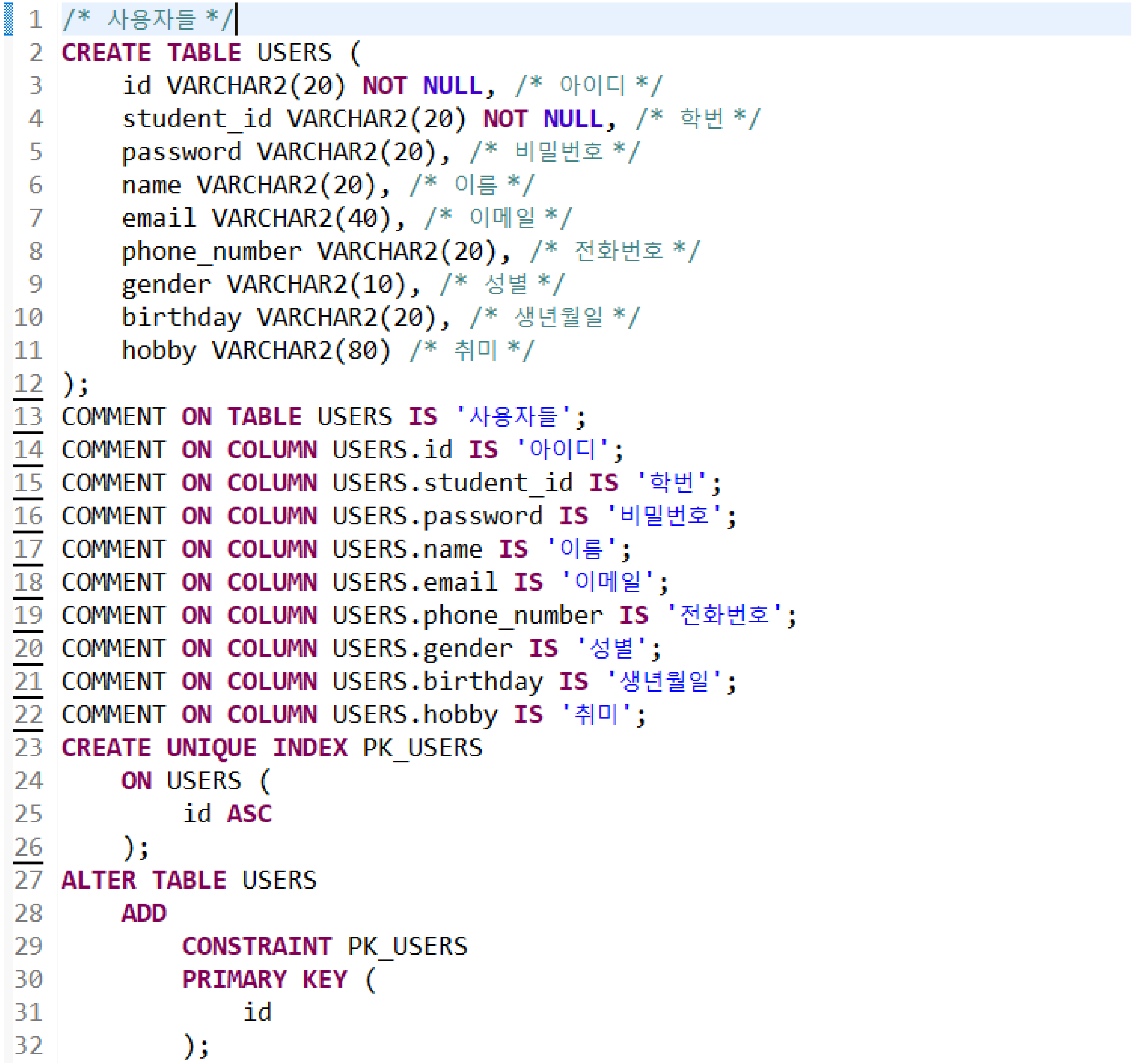
6주차 주말 과제

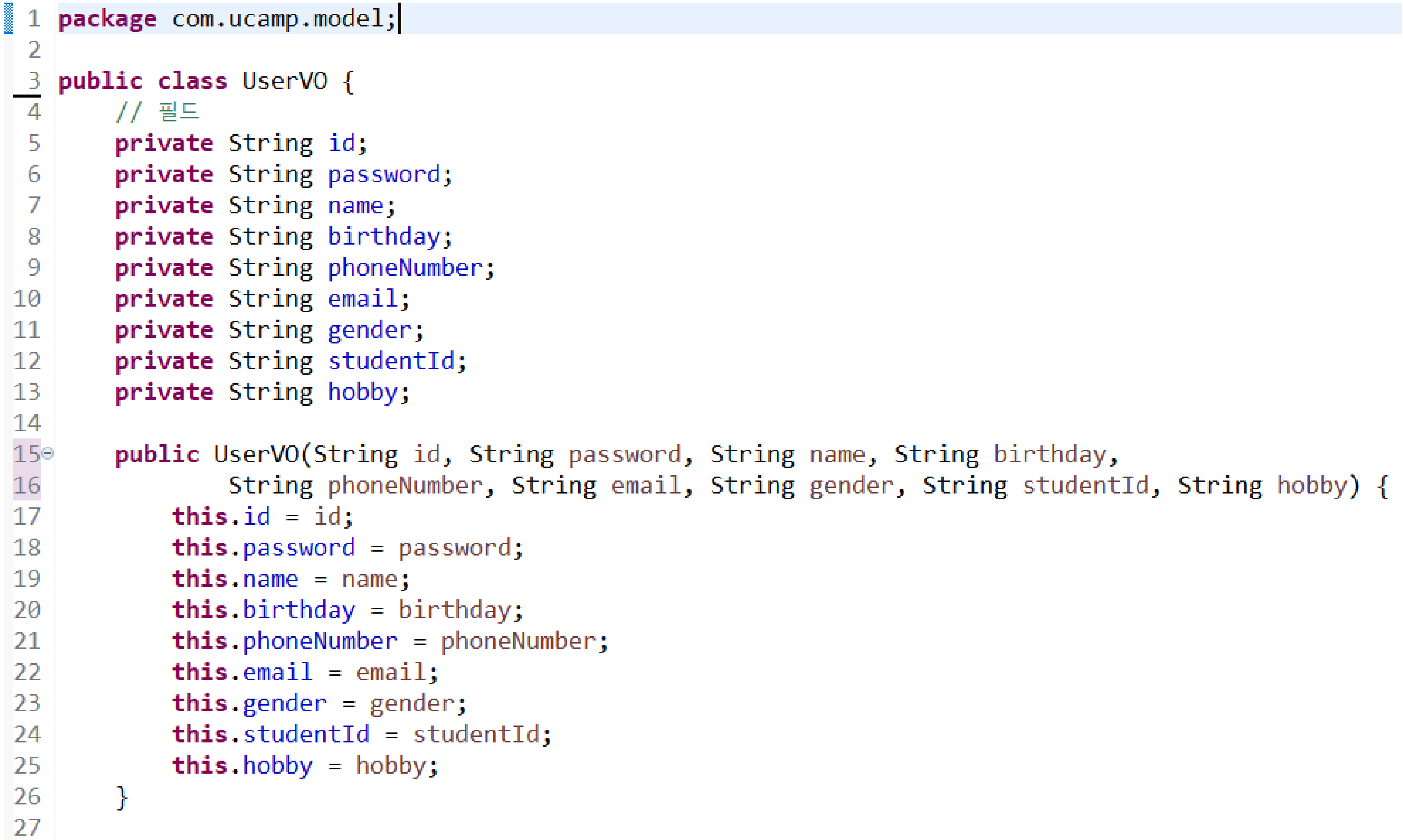
MVC model2, front controller 실습

1. Model - oracle dbms 테이블 생성



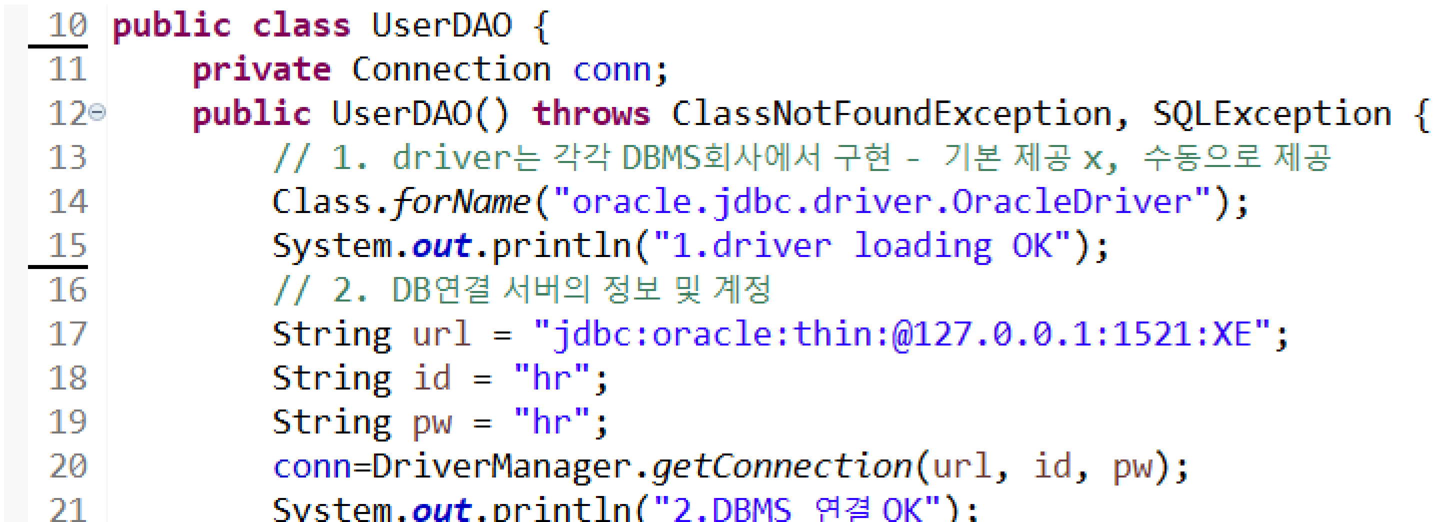
```
/* 사용자들 */
CREATE TABLE USERS (
    id VARCHAR2(20) NOT NULL, /* 아이디 */
    student_id VARCHAR2(20) NOT NULL, /* 학번 */
    password VARCHAR2(20), /* 비밀번호 */
    name VARCHAR2(20), /* 이름 */
    email VARCHAR2(40), /* 이메일 */
    phone_number VARCHAR2(20), /* 전화번호 */
    gender VARCHAR2(10), /* 성별 */
    birthday VARCHAR2(20), /* 생년월일 */
    hobby VARCHAR2(80) /* 취미 */
);
COMMENT ON TABLE USERS IS '사용자들';
COMMENT ON COLUMN USERS.id IS '아이디';
COMMENT ON COLUMN USERS.student_id IS '학번';
COMMENT ON COLUMN USERS.password IS '비밀번호';
COMMENT ON COLUMN USERS.name IS '이름';
COMMENT ON COLUMN USERS.email IS '이메일';
COMMENT ON COLUMN USERS.phone_number IS '전화번호';
COMMENT ON COLUMN USERS.gender IS '성별';
COMMENT ON COLUMN USERS.birthday IS '생년월일';
COMMENT ON COLUMN USERS.hobby IS '취미';
CREATE UNIQUE INDEX PK_USERS
    ON USERS (
        id ASC
    );
ALTER TABLE USERS
    ADD
        CONSTRAINT PK_USERS
        PRIMARY KEY (
            id
        );
```

1. Model - UserVO.java

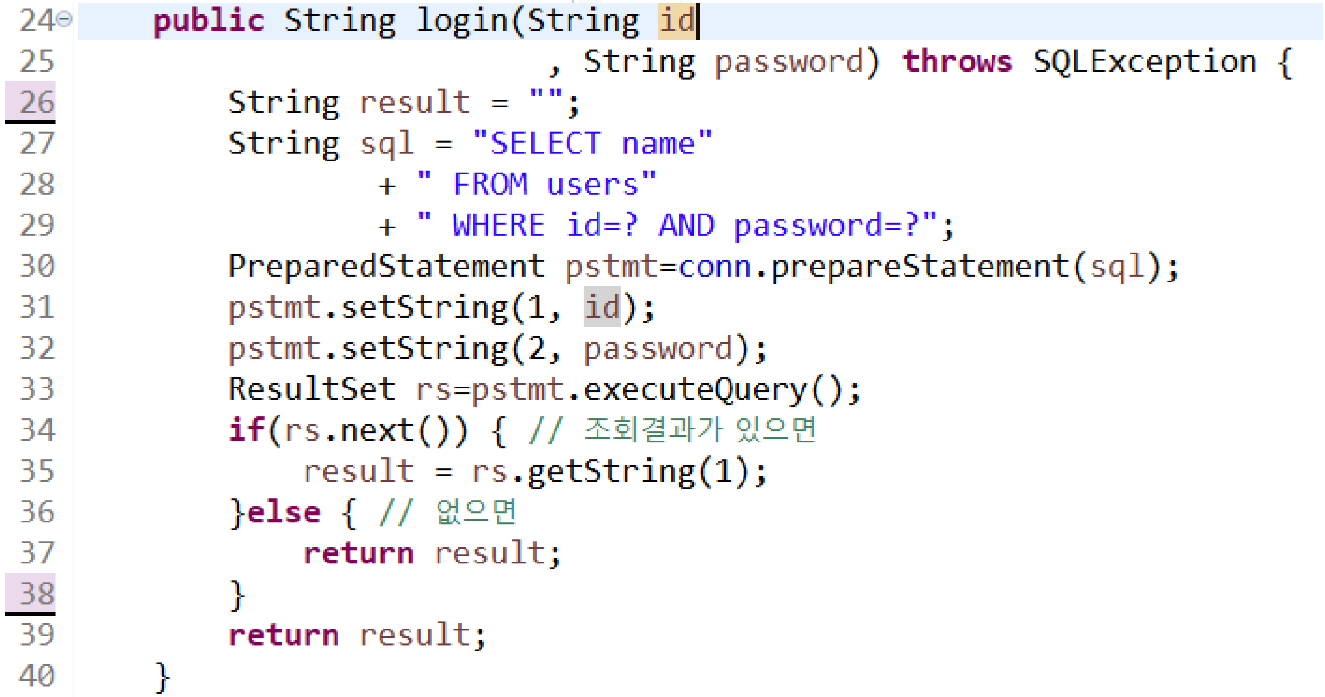


```java
package com.ucamp.model;

public class UserVO {
    // 필드
    private String id;
    private String password;
    private String name;
    private String birthday;
    private String phoneNumber;
    private String email;
    private String gender;
    private String studentId;
    private String hobby;

    public UserVO(String id, String password, String name, String birthday,
            String phoneNumber, String email, String gender, String studentId, String hobby) {
        this.id = id;
        this.password = password;
        this.name = name;
        this.birthday = birthday;
        this.phoneNumber = phoneNumber;
        this.email = email;
        this.gender = gender;
        this.studentId = studentId;
        this.hobby = hobby;
    }

```

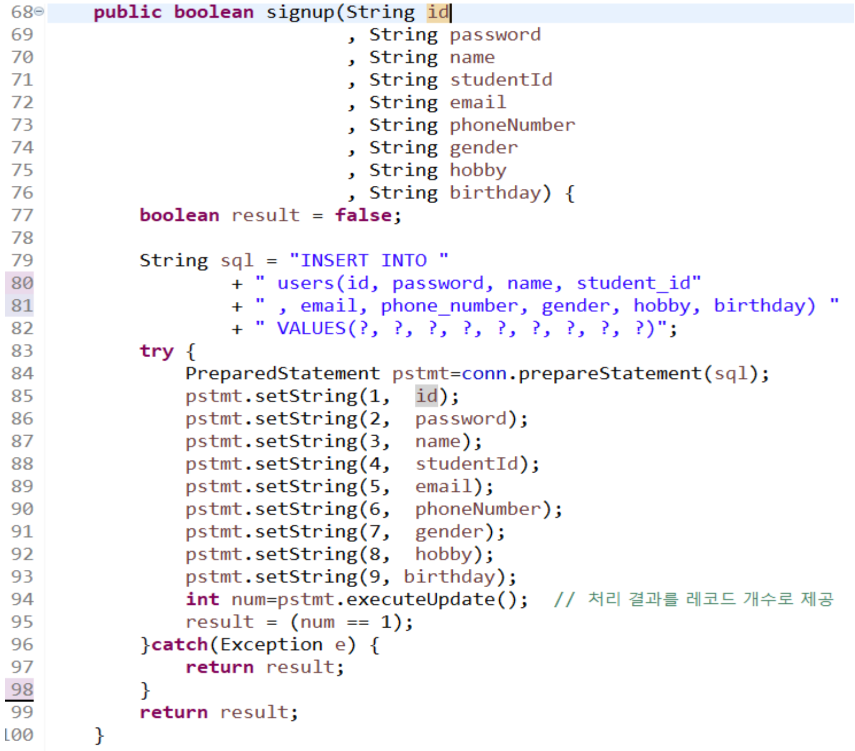
1. Model - UserDAO.java



```java
public class UserDAO {
    private Connection conn;
    public UserDAO() throws ClassNotFoundException, SQLException {
        // 1. driver는 각각 DBMS회사에서 구현 - 기본 제공 x, 수동으로 제공
        Class.forName("oracle.jdbc.driver.OracleDriver");
        System.out.println("1.driver loading OK");
        // 2. DB연결 서버의 정보 및 계정
        String url = "jdbc:oracle:thin:@127.0.0.1:1521:XE";
        String id = "hr";
        String pw = "hr";
        conn=DriverManager.getConnection(url, id, pw);
        System.out.println("2.DBMS 연결 OK");
    }
```

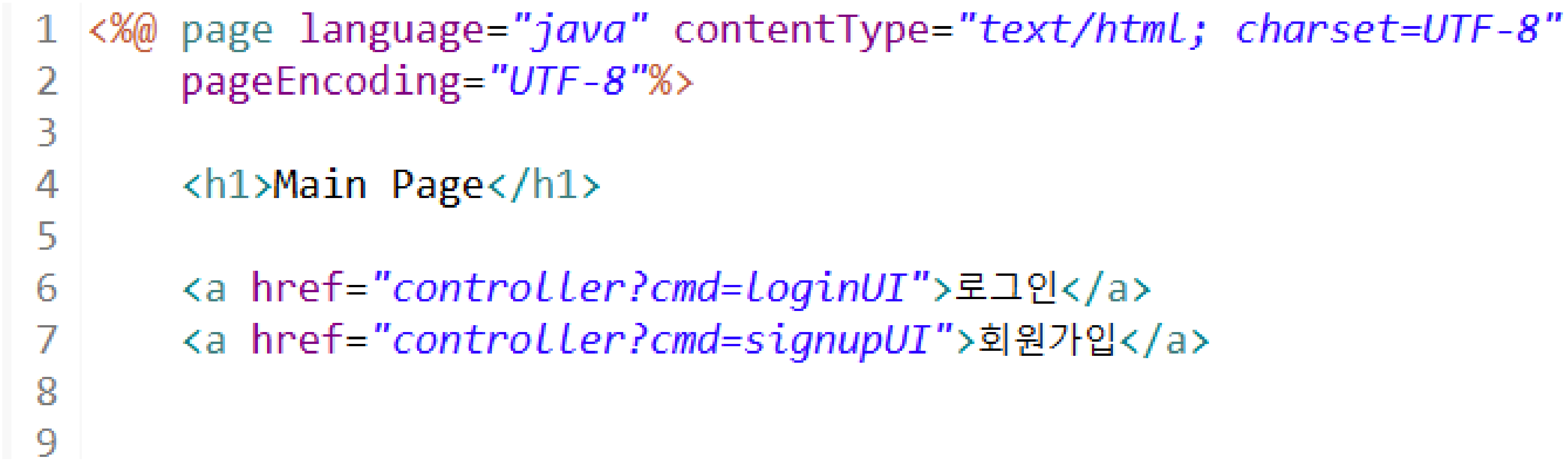


```java
    public String login(String id
                        , String password) throws SQLException {
        String result = "";
        String sql = "SELECT name"
                + " FROM users"
                + " WHERE id=? AND password=?";
        PreparedStatement pstmt=conn.prepareStatement(sql);
        pstmt.setString(1, id);
        pstmt.setString(2, password);
        ResultSet rs=pstmt.executeQuery();
        if(rs.next()) { // 조회결과가 있으면
            result = rs.getString(1);
        }else { // 없으면
            return result;
        }
        return result;
    }
```



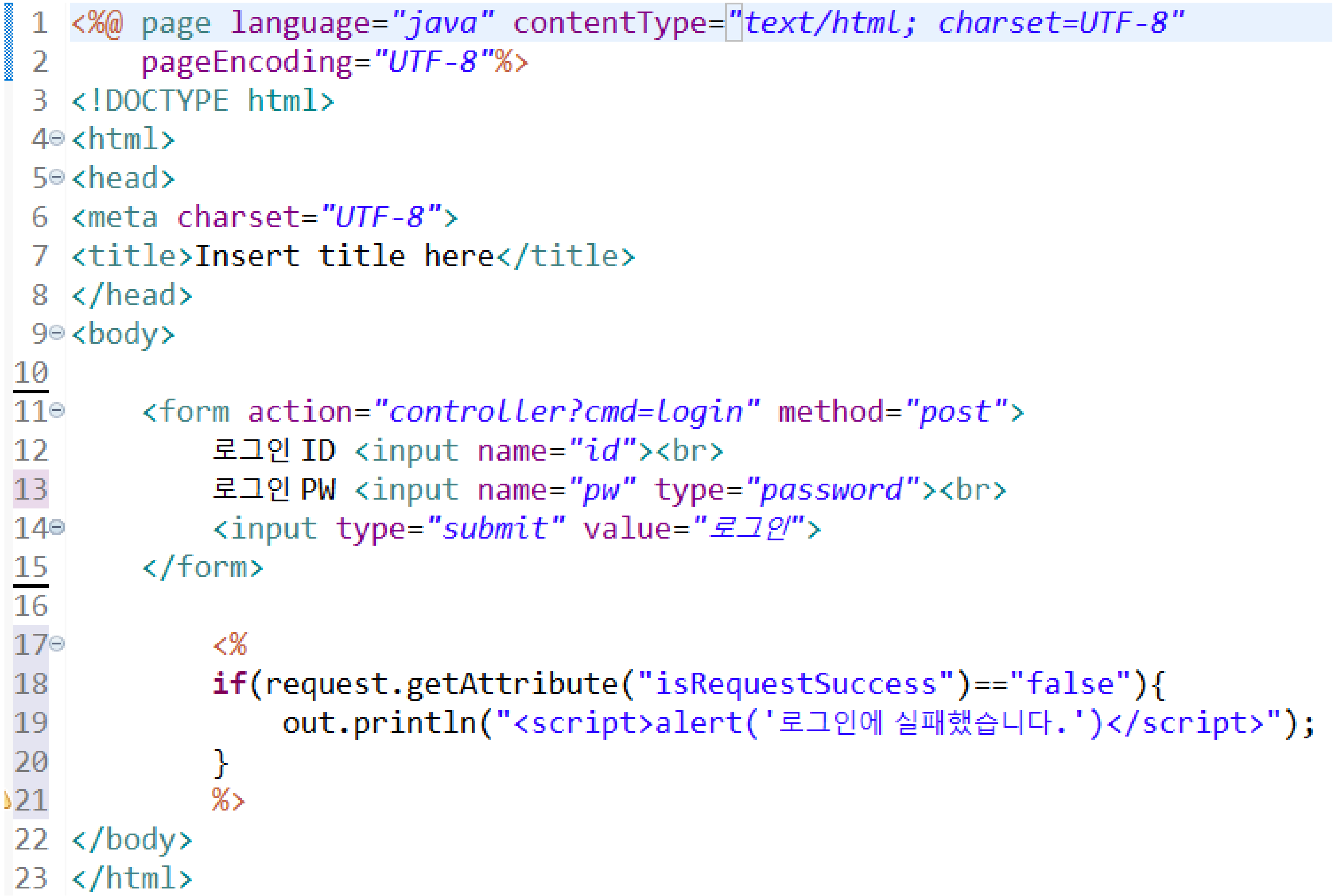
```java
    public boolean signup(String id
                            , String password
                            , String name
                            , String studentId
                            , String email
                            , String phoneNumber
                            , String gender
                            , String hobby
                            , String birthday) {
        boolean result = false;

        String sql = "INSERT INTO "
                + " users(id, password, name, student_id"
                + " , email, phone_number, gender, hobby, birthday) "
                + " VALUES(?, ?, ?, ?, ?, ?, ?, ?, ?)";
        try {
            PreparedStatement pstmt=conn.prepareStatement(sql);
            pstmt.setString(1,  id);
            pstmt.setString(2,  password);
            pstmt.setString(3,  name);
            pstmt.setString(4,  studentId);
            pstmt.setString(5,  email);
            pstmt.setString(6,  phoneNumber);
            pstmt.setString(7,  gender);
            pstmt.setString(8,  hobby);
            pstmt.setString(9, birthday);
            int num=pstmt.executeUpdate();  // 처리 결과를 레코드 개수로 제공
            result = (num == 1);
        }catch(Exception e) {
            return result;
        }
        return result;
    }
```

2. View - main.jsp



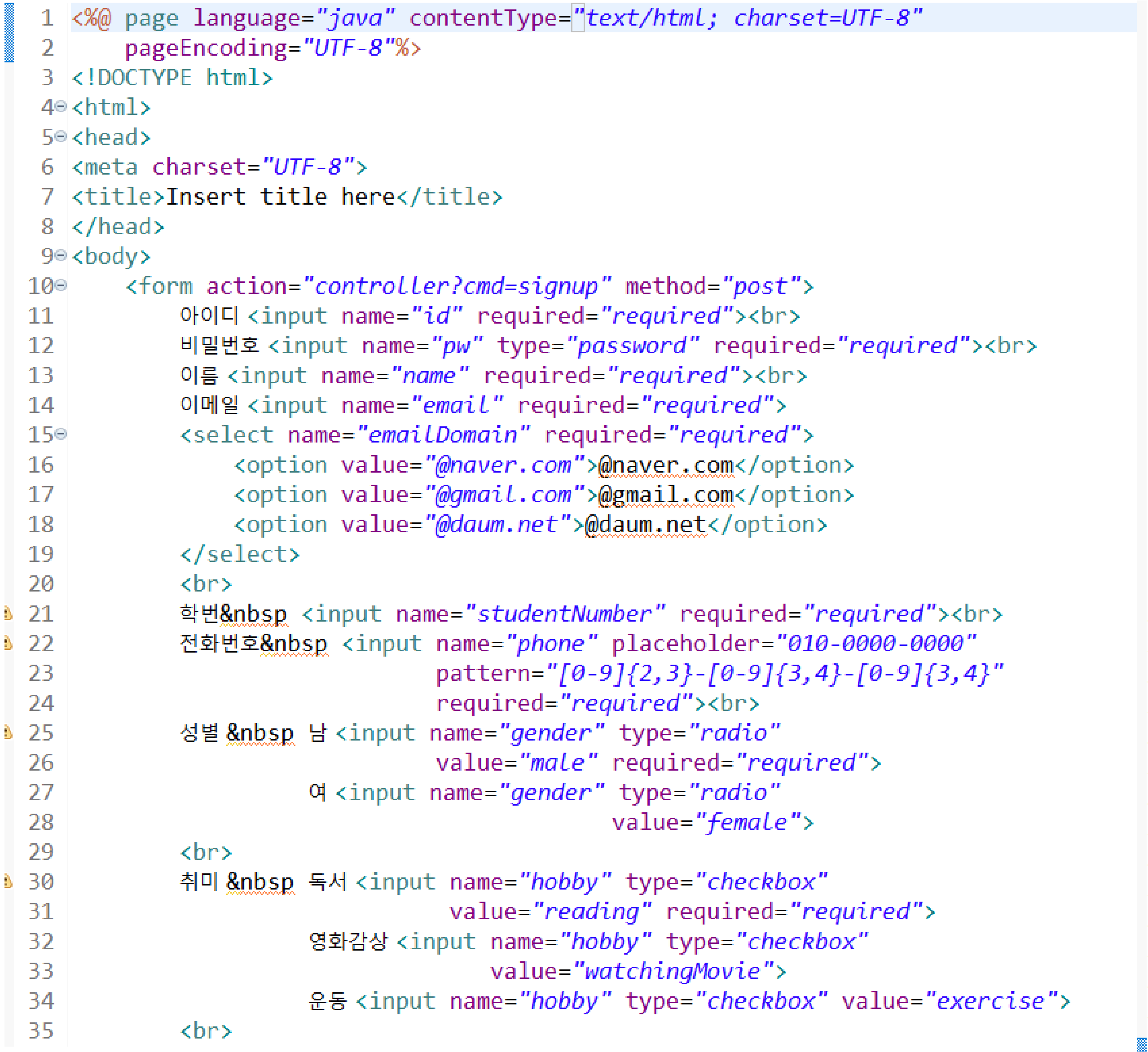
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>

    <h1>Main Page</h1>

    <a href="controller?cmd=loginUI">로그인</a>
    <a href="controller?cmd=signupUI">회원가입</a>


```

2. View - login.jsp

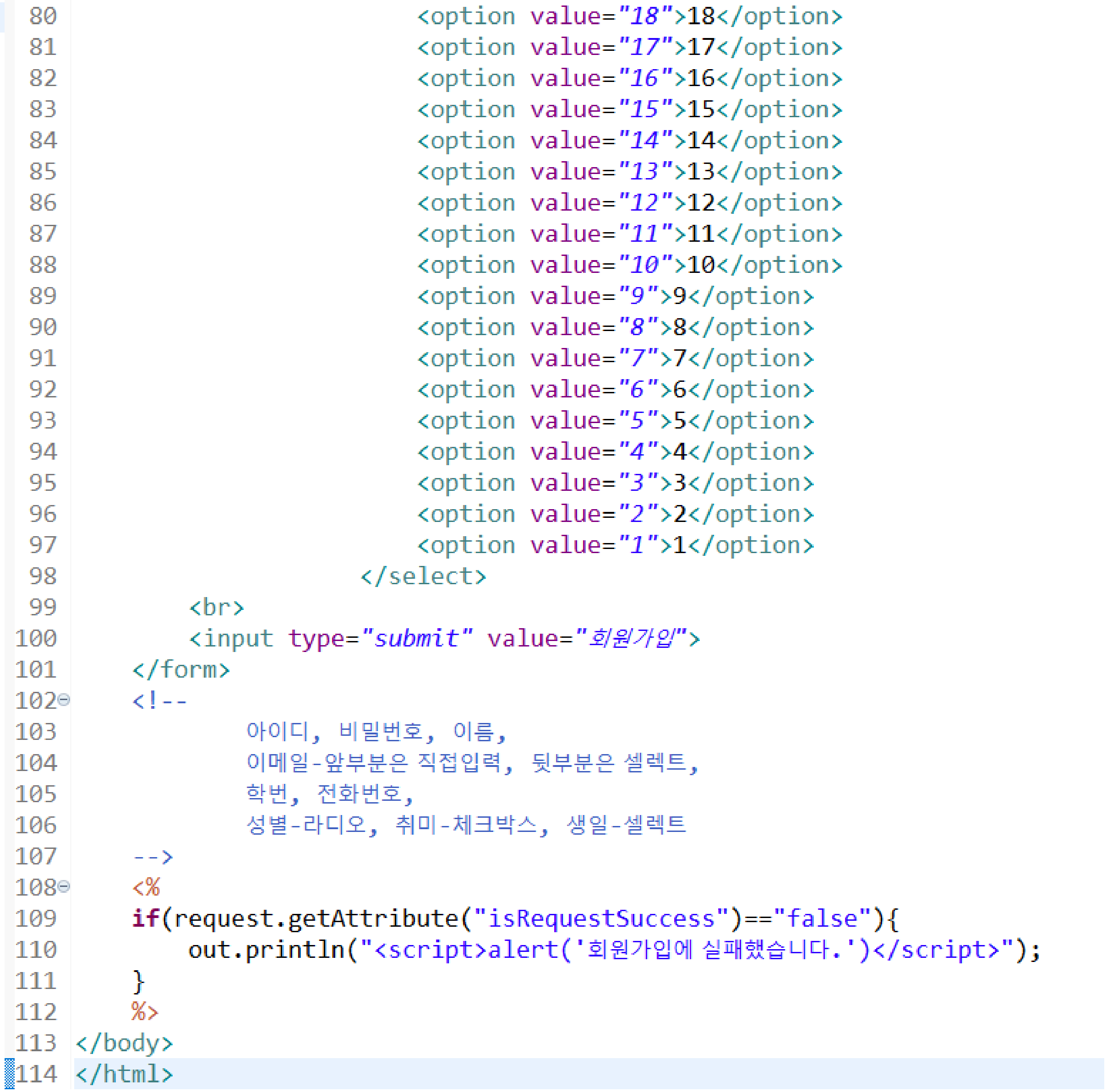


```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>

    <form action="controller?cmd=login" method="post">
        로그인 ID <input name="id"><br>
        로그인 PW <input name="pw" type="password"><br>
        <input type="submit" value="로그인">
    </form>

        <%
        if(request.getAttribute("isRequestSuccess")=="false"){
            out.println("<script>alert('로그인에 실패했습니다.')</script>");
        }
        %>
</body>
</html>
```

2. View - signup.jsp

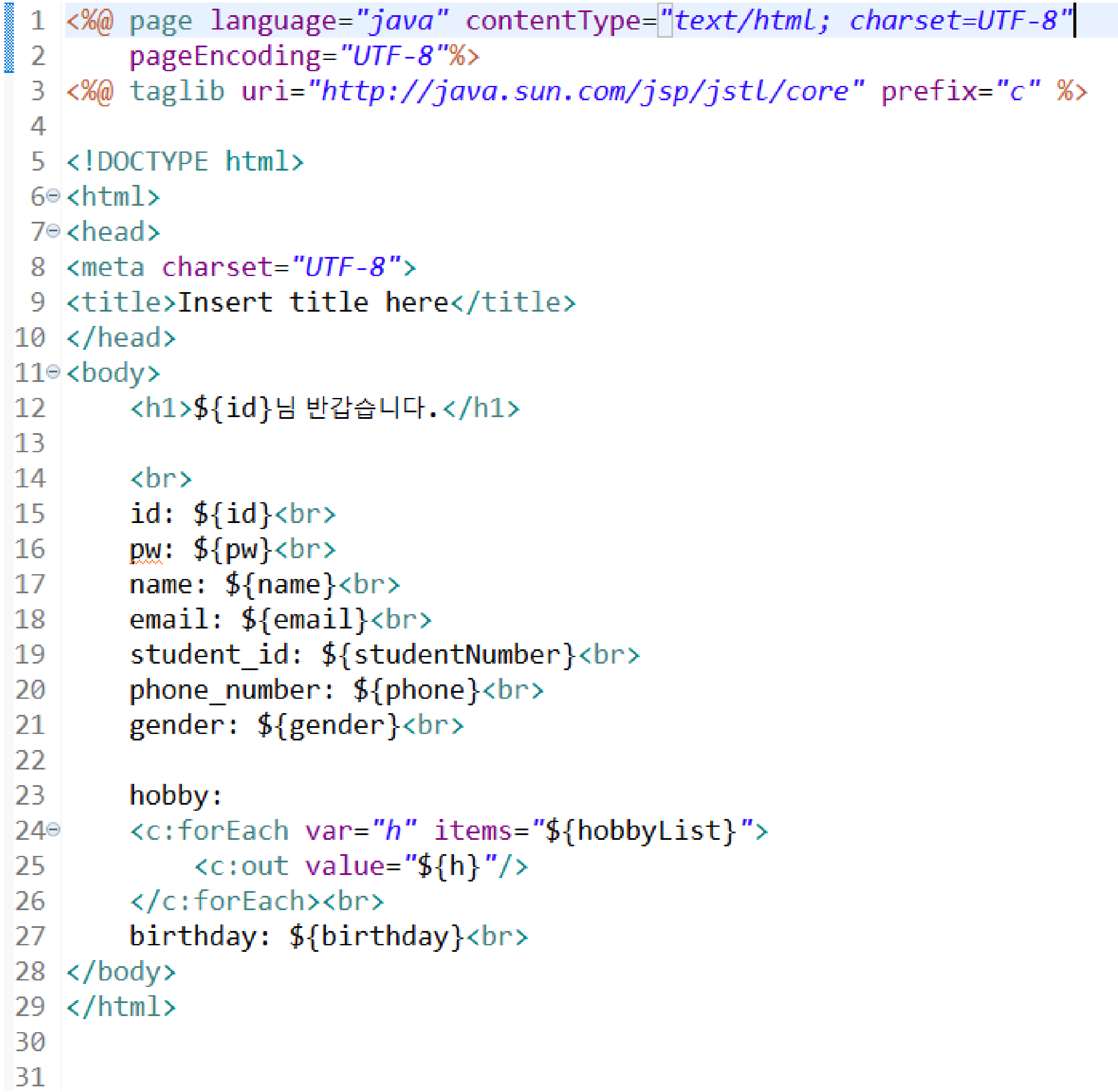


```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
    <form action="controller?cmd=signup" method="post">
        아이디 <input name="id" required="required"><br>
        비밀번호 <input name="pw" type="password" required="required"><br>
        이름 <input name="name" required="required"><br>
        이메일 <input name="email" required="required">
        <select name="emailDomain" required="required">
            <option value="@naver.com">@naver.com</option>
            <option value="@gmail.com">@gmail.com</option>
            <option value="@daum.net">@daum.net</option>
        </select>
        <br>
        학번&nbsp <input name="studentNumber" required="required"><br>
        전화번호&nbsp <input name="phone" placeholder="010-0000-0000"
                           pattern="[0-9]{2,3}-[0-9]{3,4}-[0-9]{3,4}"
                           required="required"><br>
        성별 &nbsp 남 <input name="gender" type="radio"
                           value="male" required="required">
                 여 <input name="gender" type="radio"
                                        value="female">
        <br>
        취미 &nbsp 독서 <input name="hobby" type="checkbox"
                            value="reading" required="required">
                 영화감상 <input name="hobby" type="checkbox"
                               value="watchingMovie">
                 운동 <input name="hobby" type="checkbox" value="exercise">
        <br>
```



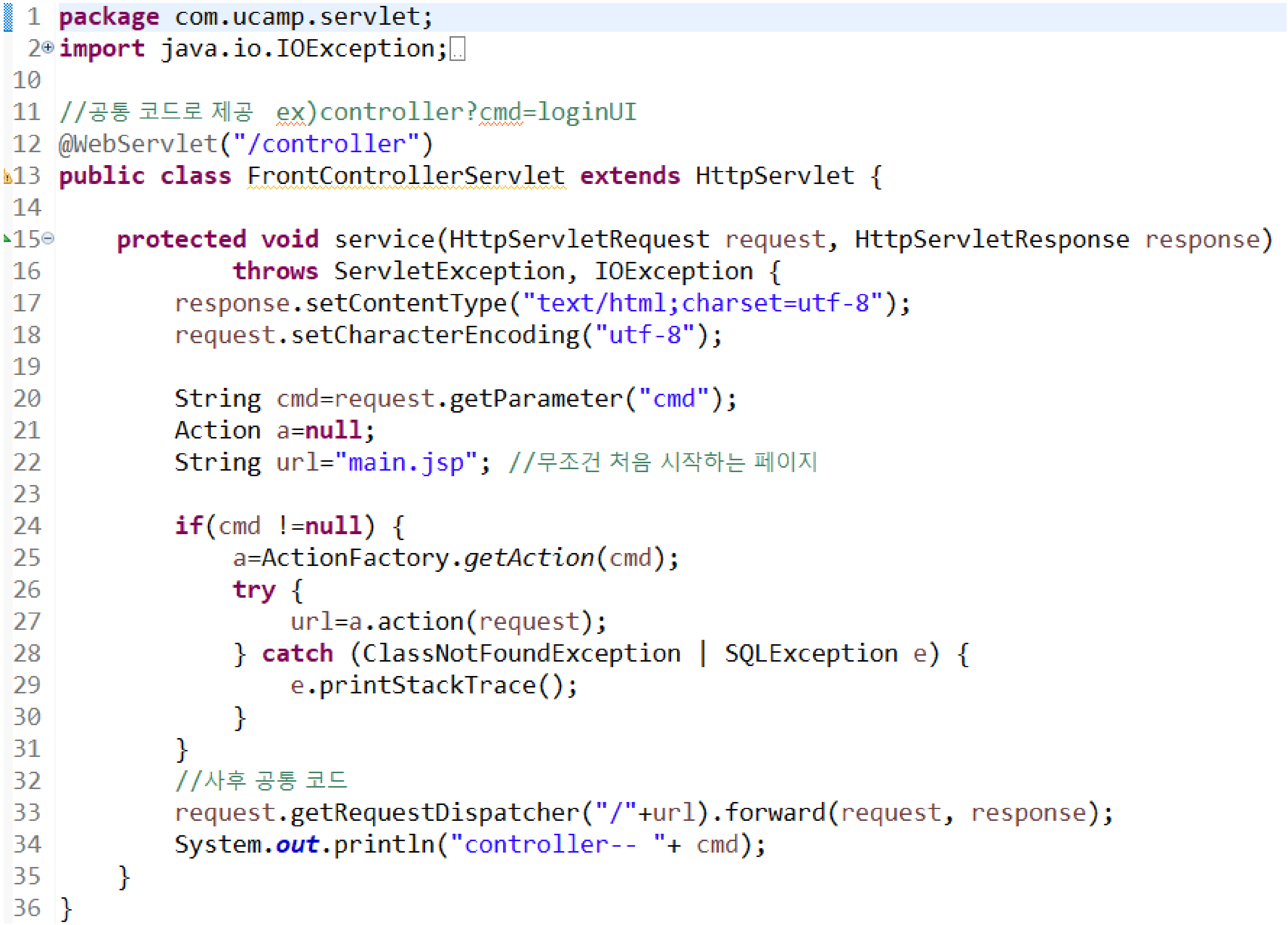
```
                        <option value="18">18</option>
                        <option value="17">17</option>
                        <option value="16">16</option>
                        <option value="15">15</option>
                        <option value="14">14</option>
                        <option value="13">13</option>
                        <option value="12">12</option>
                        <option value="11">11</option>
                        <option value="10">10</option>
                        <option value="9">9</option>
                        <option value="8">8</option>
                        <option value="7">7</option>
                        <option value="6">6</option>
                        <option value="5">5</option>
                        <option value="4">4</option>
                        <option value="3">3</option>
                        <option value="2">2</option>
                        <option value="1">1</option>
                    </select>
        <br>
        <input type="submit" value="회원가입">
    </form>
    <!--
            아이디, 비밀번호, 이름,
            이메일-앞부분은 직접입력, 뒷부분은 셀렉트,
            학번, 전화번호,
            성별-라디오, 취미-체크박스, 생일-셀렉트
    -->
    <%
    if(request.getAttribute("isRequestSuccess")=="false"){
        out.println("<script>alert('회원가입에 실패했습니다.')</script>");
    }
    %>
</body>
</html>
```

2. View - userInfo.jsp



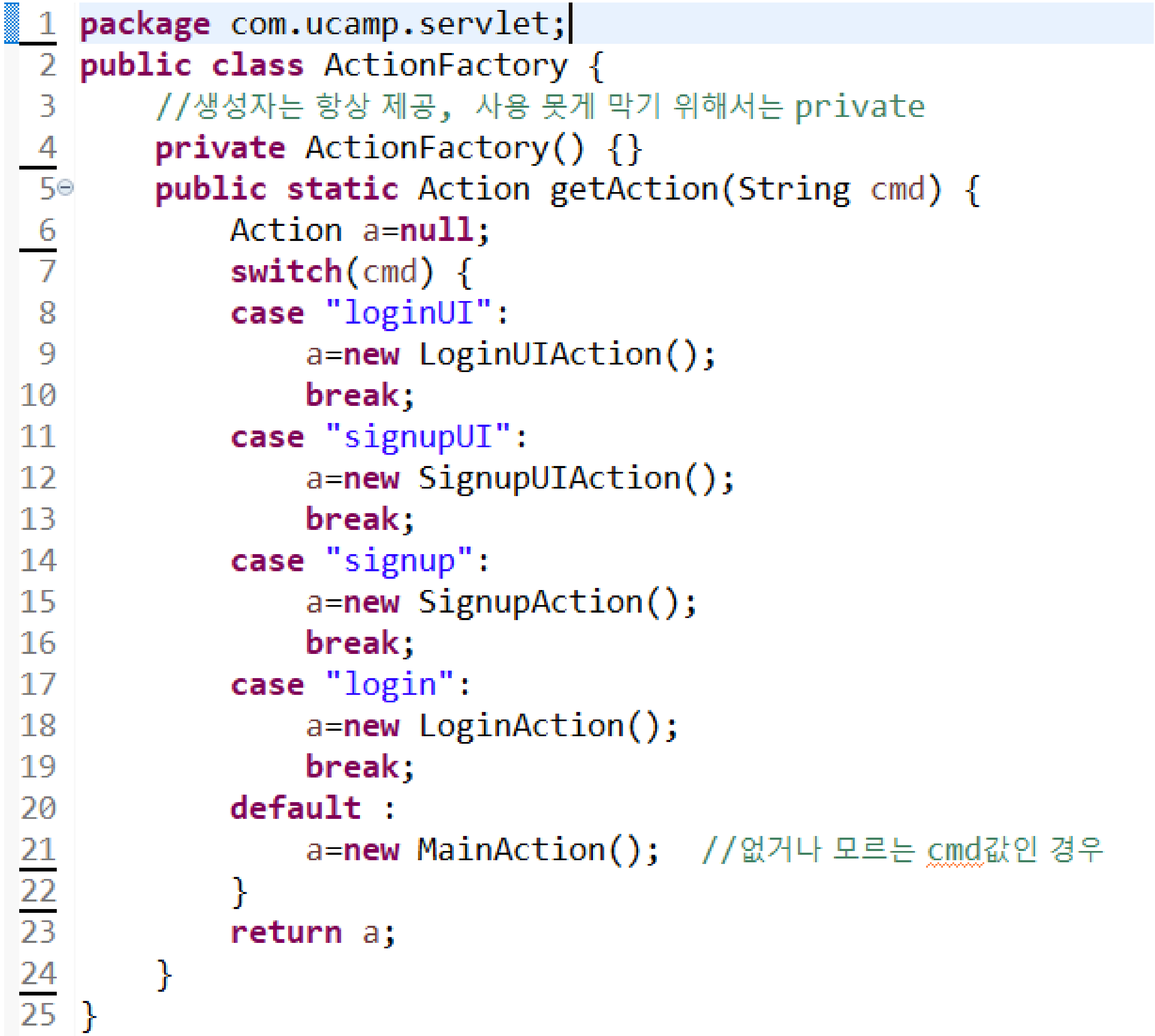
```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
    <h1>${id}님 반갑습니다.</h1>

    <br>
    id: ${id}<br>
    pw: ${pw}<br>
    name: ${name}<br>
    email: ${email}<br>
    student_id: ${studentNumber}<br>
    phone_number: ${phone}<br>
    gender: ${gender}<br>

    hobby:
    <c:forEach var="h" items="${hobbyList}">
        <c:out value="${h}"/>
    </c:forEach><br>
    birthday: ${birthday}<br>
</body>
</html>


```

3. Controller - FrontControllerServlet.java



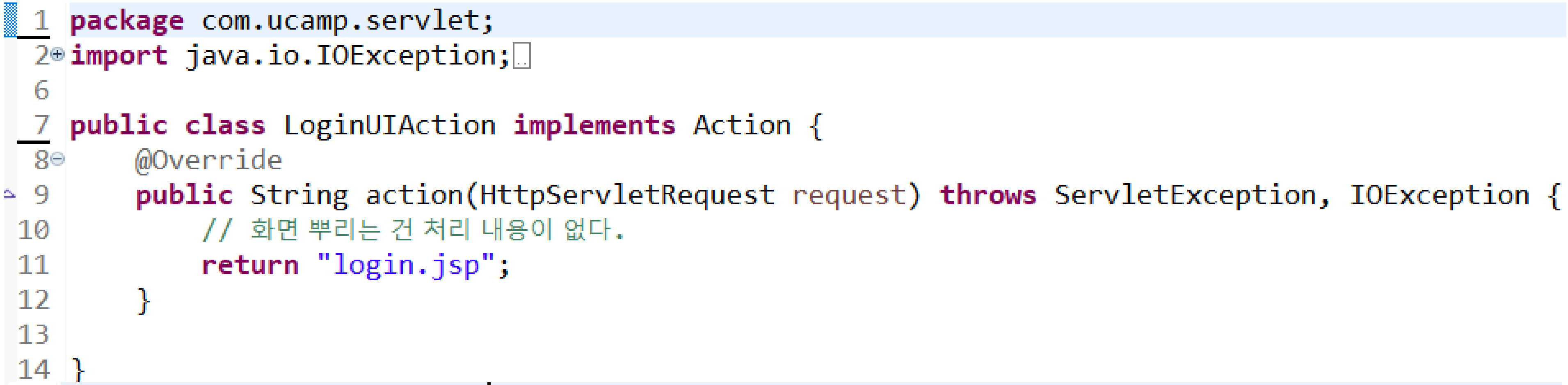
```java
package com.ucamp.servlet;
import java.io.IOException;

//공통 코드로 제공  ex)controller?cmd=loginUI
@WebServlet("/controller")
public class FrontControllerServlet extends HttpServlet {

    protected void service(HttpServletRequest request, HttpServletResponse response)
            throws ServletException, IOException {
        response.setContentType("text/html;charset=utf-8");
        request.setCharacterEncoding("utf-8");

        String cmd=request.getParameter("cmd");
        Action a=null;
        String url="main.jsp"; //무조건 처음 시작하는 페이지

        if(cmd !=null) {
            a=ActionFactory.getAction(cmd);
            try {
                url=a.action(request);
            } catch (ClassNotFoundException | SQLException e) {
                e.printStackTrace();
            }
        }
        //사후 공통 코드
        request.getRequestDispatcher("/"+url).forward(request, response);
        System.out.println("controller-- "+ cmd);
    }
}
```

3. Controller - ActionFactory.java

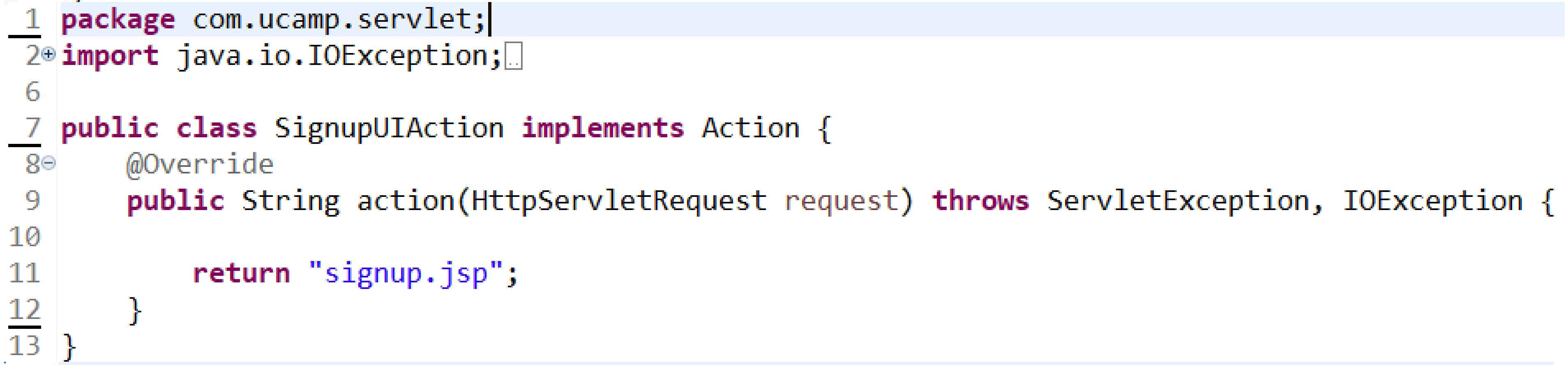


```java
package com.ucamp.servlet;
public class ActionFactory {
    //생성자는 항상 제공, 사용 못게 막기 위해서는 private
    private ActionFactory() {}
    public static Action getAction(String cmd) {
        Action a=null;
        switch(cmd) {
        case "loginUI":
            a=new LoginUIAction();
            break;
        case "signupUI":
            a=new SignupUIAction();
            break;
        case "signup":
            a=new SignupAction();
            break;
        case "login":
            a=new LoginAction();
            break;
        default :
            a=new MainAction();   //없거나 모르는 cmd값인 경우
        }
        return a;
    }
}
```

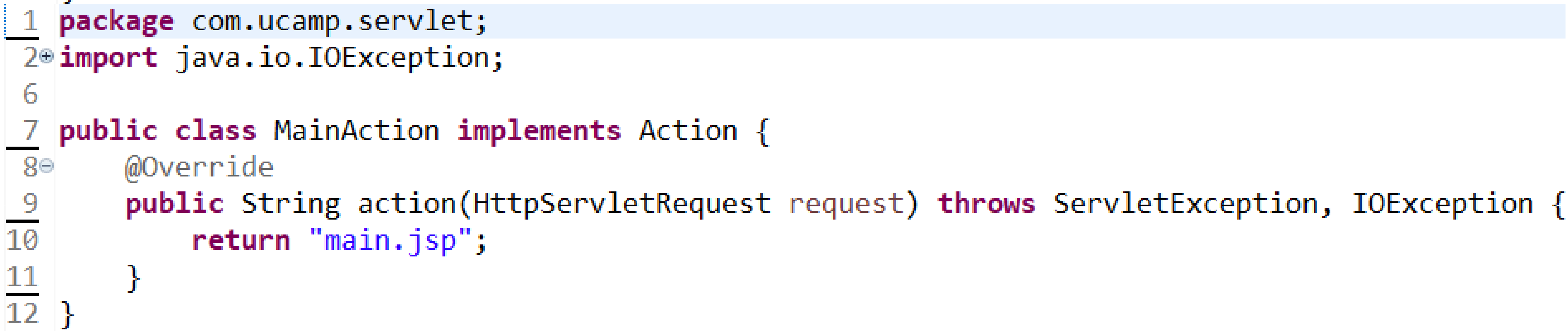
3. Controller - LoginUIAction.java, SignupUIAction.java, MainAction.java



```java
package com.ucamp.servlet;
import java.io.IOException;

public class LoginUIAction implements Action {
    @Override
    public String action(HttpServletRequest request) throws ServletException, IOException {
        // 화면 뿌리는 건 처리 내용이 없다.
        return "login.jsp";
    }

}
```

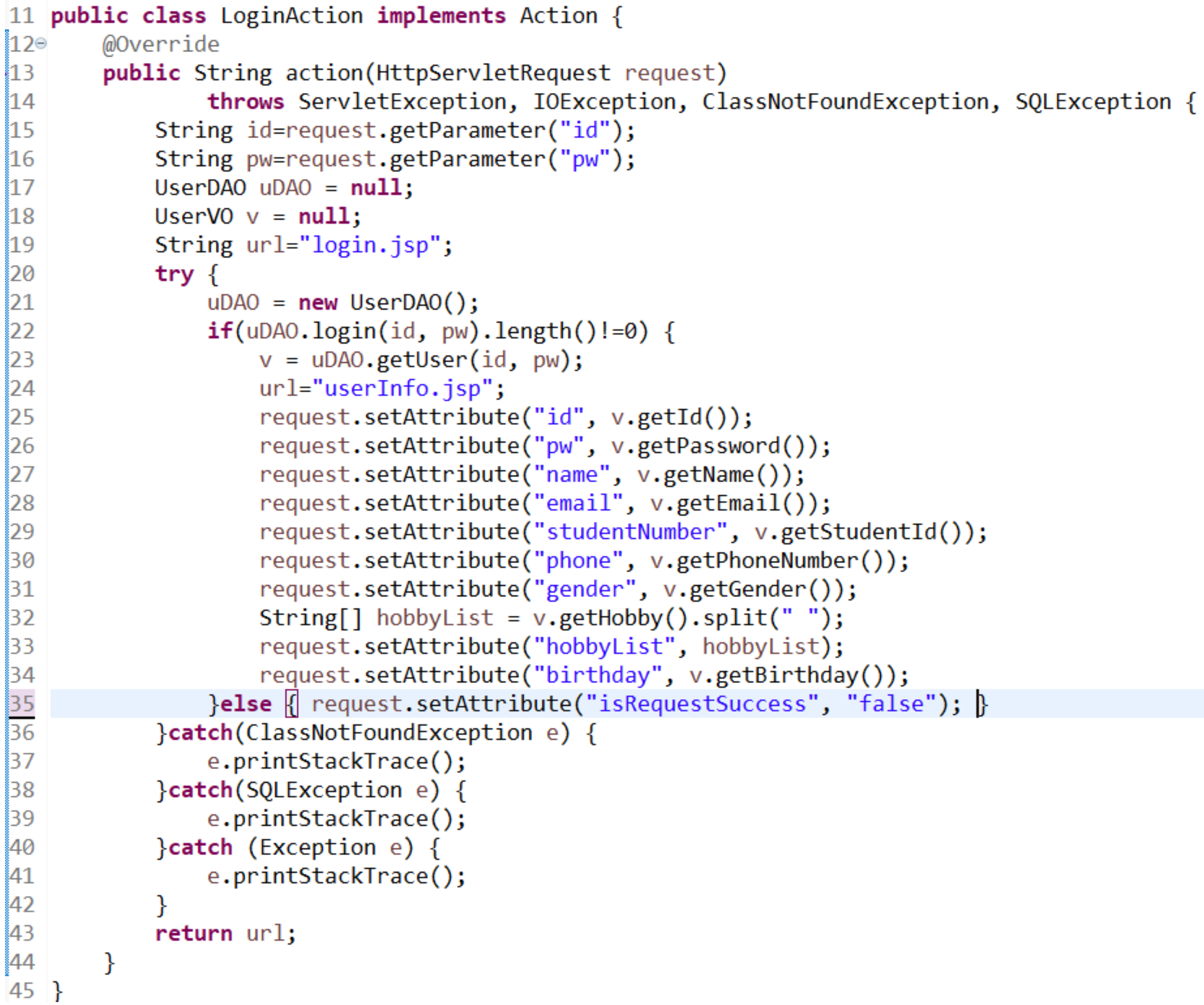


```java
package com.ucamp.servlet;
import java.io.IOException;

public class SignupUIAction implements Action {
    @Override
    public String action(HttpServletRequest request) throws ServletException, IOException {

        return "signup.jsp";
    }
}
```

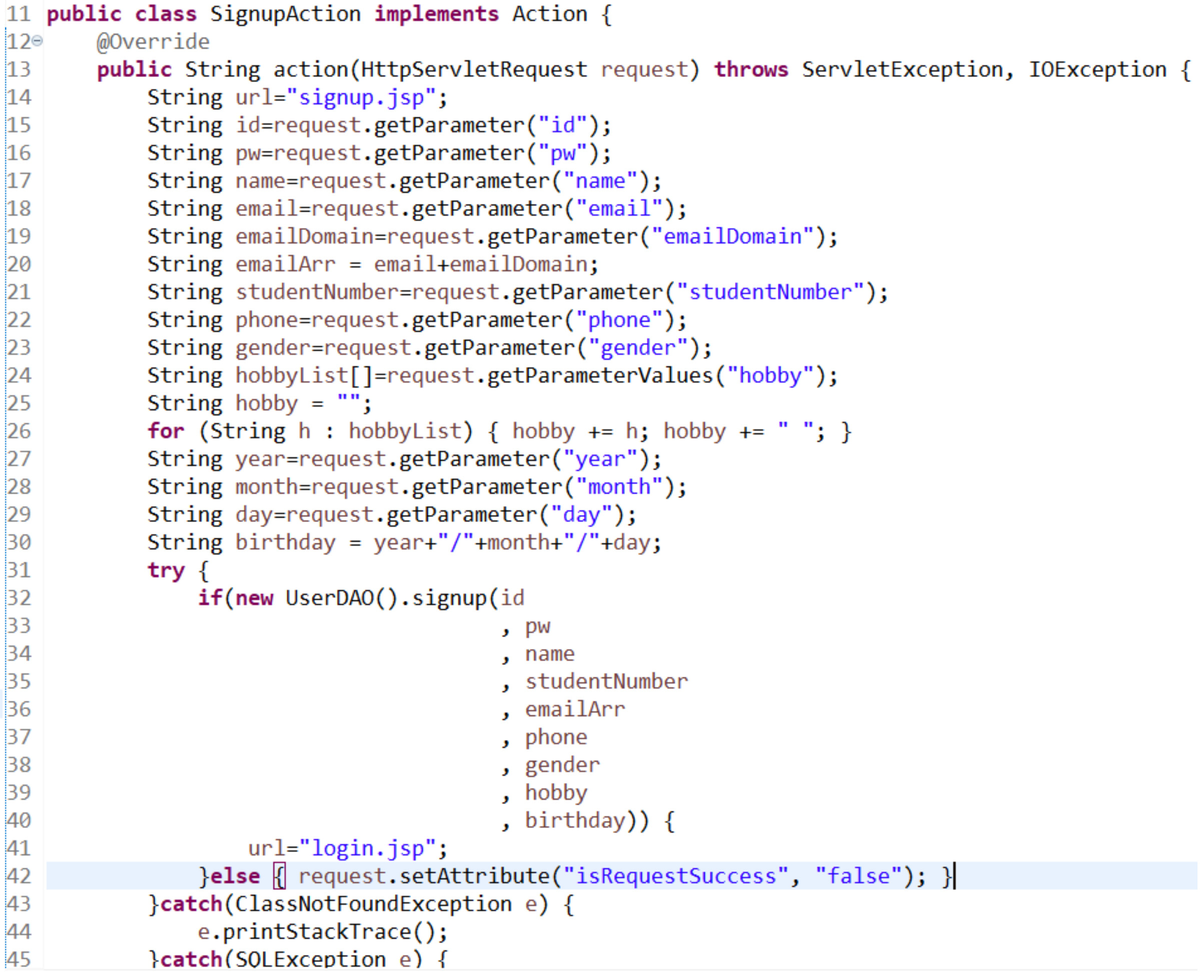


```java
package com.ucamp.servlet;
import java.io.IOException;

public class MainAction implements Action {
    @Override
    public String action(HttpServletRequest request) throws ServletException, IOException {
        return "main.jsp";
    }
}
```

3. Controller - LoginAction.java, SignupAction.java



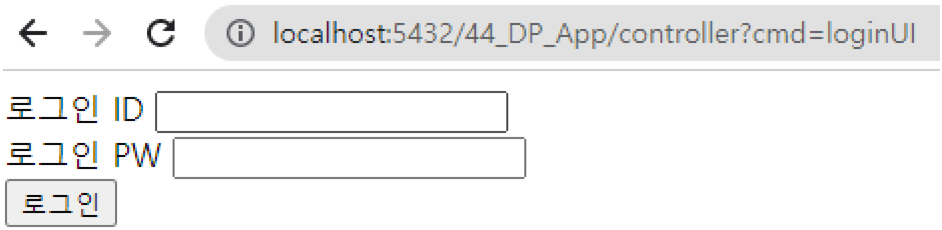
```java
public class LoginAction implements Action {
    @Override
    public String action(HttpServletRequest request)
            throws ServletException, IOException, ClassNotFoundException, SQLException {
        String id=request.getParameter("id");
        String pw=request.getParameter("pw");
        UserDAO uDAO = null;
        UserVO v = null;
        String url="login.jsp";
        try {
            uDAO = new UserDAO();
            if(uDAO.login(id, pw).length()!=0) {
                v = uDAO.getUser(id, pw);
                url="userInfo.jsp";
                request.setAttribute("id", v.getId());
                request.setAttribute("pw", v.getPassword());
                request.setAttribute("name", v.getName());
                request.setAttribute("email", v.getEmail());
                request.setAttribute("studentNumber", v.getStudentId());
                request.setAttribute("phone", v.getPhoneNumber());
                request.setAttribute("gender", v.getGender());
                String[] hobbyList = v.getHobby().split(" ");
                request.setAttribute("hobbyList", hobbyList);
                request.setAttribute("birthday", v.getBirthday());
            }else { request.setAttribute("isRequestSuccess", "false"); }
        }catch(ClassNotFoundException e) {
            e.printStackTrace();
        }catch(SQLException e) {
            e.printStackTrace();
        }catch (Exception e) {
            e.printStackTrace();
        }
        return url;
    }
}
```



```java
public class SignupAction implements Action {
    @Override
    public String action(HttpServletRequest request) throws ServletException, IOException {
        String url="signup.jsp";
        String id=request.getParameter("id");
        String pw=request.getParameter("pw");
        String name=request.getParameter("name");
        String email=request.getParameter("email");
        String emailDomain=request.getParameter("emailDomain");
        String emailArr = email+emailDomain;
        String studentNumber=request.getParameter("studentNumber");
        String phone=request.getParameter("phone");
        String gender=request.getParameter("gender");
        String hobbyList[]=request.getParameterValues("hobby");
        String hobby = "";
        for (String h : hobbyList) { hobby += h; hobby += " "; }
        String year=request.getParameter("year");
        String month=request.getParameter("month");
        String day=request.getParameter("day");
        String birthday = year+"/"+month+"/"+day;
        try {
            if(new UserDAO().signup(id
                                    , pw
                                    , name
                                    , studentNumber
                                    , emailArr
                                    , phone
                                    , gender
                                    , hobby
                                    , birthday)) {
                url="login.jsp";
            }else { request.setAttribute("isRequestSuccess", "false"); }
        }catch(ClassNotFoundException e) {
            e.printStackTrace();
        }catch(SQLException e) {
```

4. 실행

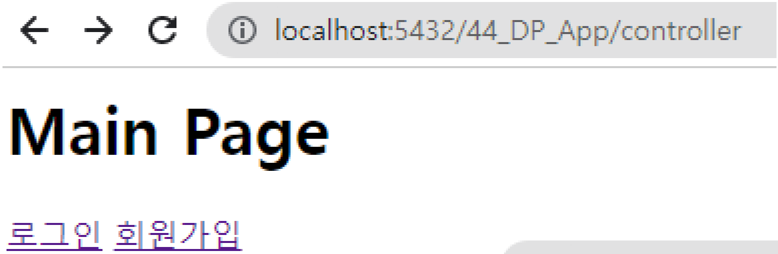
로그인성공



로그인 ID

로그인 PW

request : LoginUI

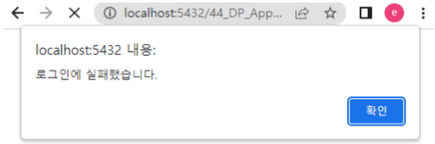


Main Page

로그인 회원가입

로그인실패

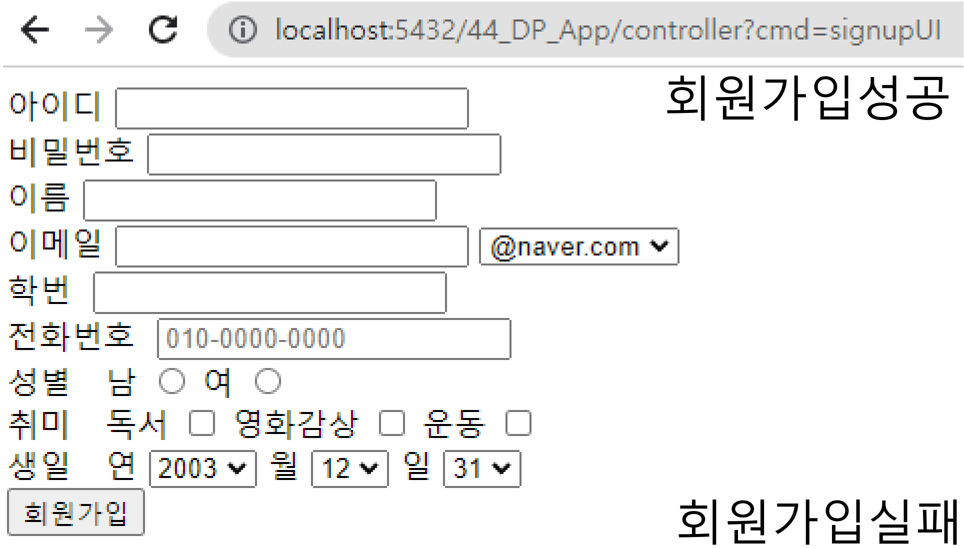
response
: login.jsp



localhost:5432 내용:

로그인에 실패했습니다.

request
: SignupUI



아이디

비밀번호

이름

이메일 @naver.com

학번

전화번호 010-0000-0000

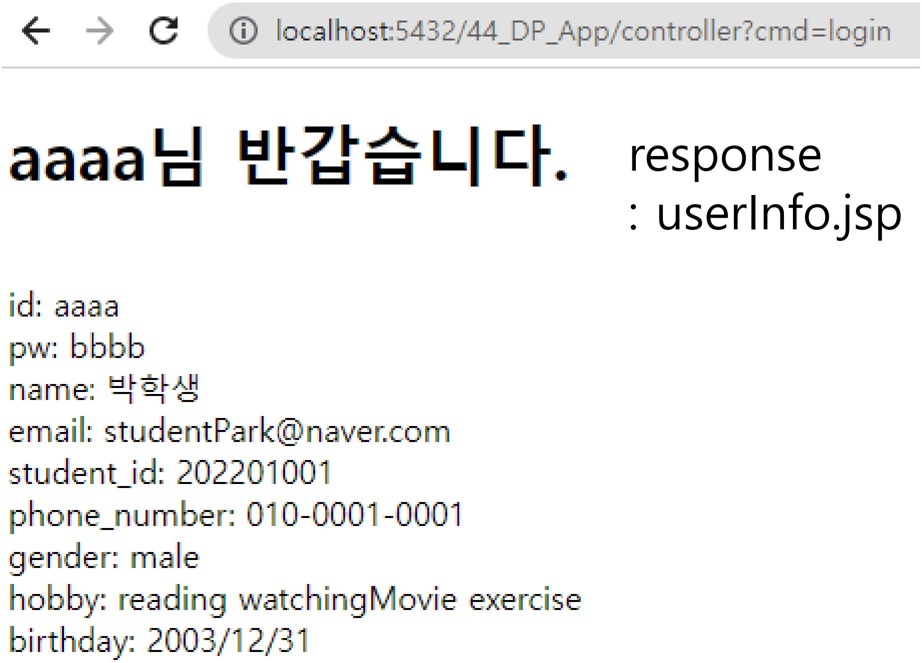
성별 남 ○ 여 ○

취미 독서 □ 영화감상 □ 운동 □

생일 연 2003 월 12 일 31

회원가입성공

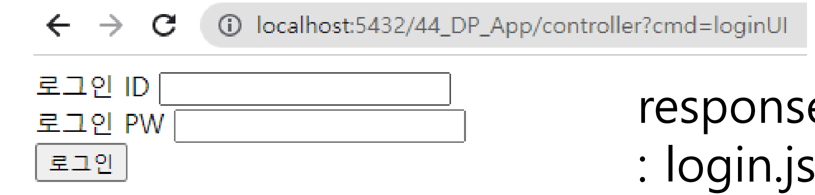
회원가입실패



aaaa님 반갑습니다.

response
: userInfo.jsp

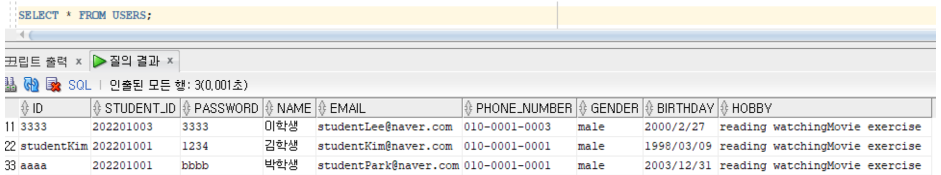
id: aaaa
pw: bbbb
name: 박학생
email: studentPark@naver.com
student_id: 202201001
phone_number: 010-0001-0001
gender: male
hobby: reading watchingMovie exercise
birthday: 2003/12/31



로그인 ID

로그인 PW

response
: login.jsp

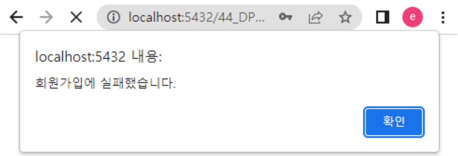


```
SELECT * FROM USERS;
```

인출된 모든 행: 3(0.001초)

	ID	STUDENT_ID	PASSWORD	NAME	EMAIL	PHONE_NUMBER	GENDER	BIRTHDAY	HOBBY
1	3333	202201003	3333	이학생	studentLee@naver.com	010-0001-0003	male	2000/2/27	reading watchingMovie exercise
2	studentKim	202201001	1234	김학생	studentKim@naver.com	010-0001-0001	male	1998/03/09	reading watchingMovie exercise
3	aaaa	202201001	bbbb	박학생	studentPark@naver.com	010-0001-0001	male	2003/12/31	reading watchingMovie exercise

response
: signup.jsp



localhost:5432 내용:

회원가입에 실패했습니다.