

서블릿으로 조회 구현

# 조회하기

## 1. UserVO 수정

```
20     }
27     public String getPwd() {
28         return pwd;
29     }
30     public void setPwd(String pwd) {
31         this.pwd = pwd;
32     }
33     public UserVO() {
34     }
35     public UserVO(int no, String name, String email, String pwd) {
36         this.no=no;
37         this.name=name;
38         this.email=email;
39         this.pwd=pwd; }
40     }
41 }
```

# 조회하기

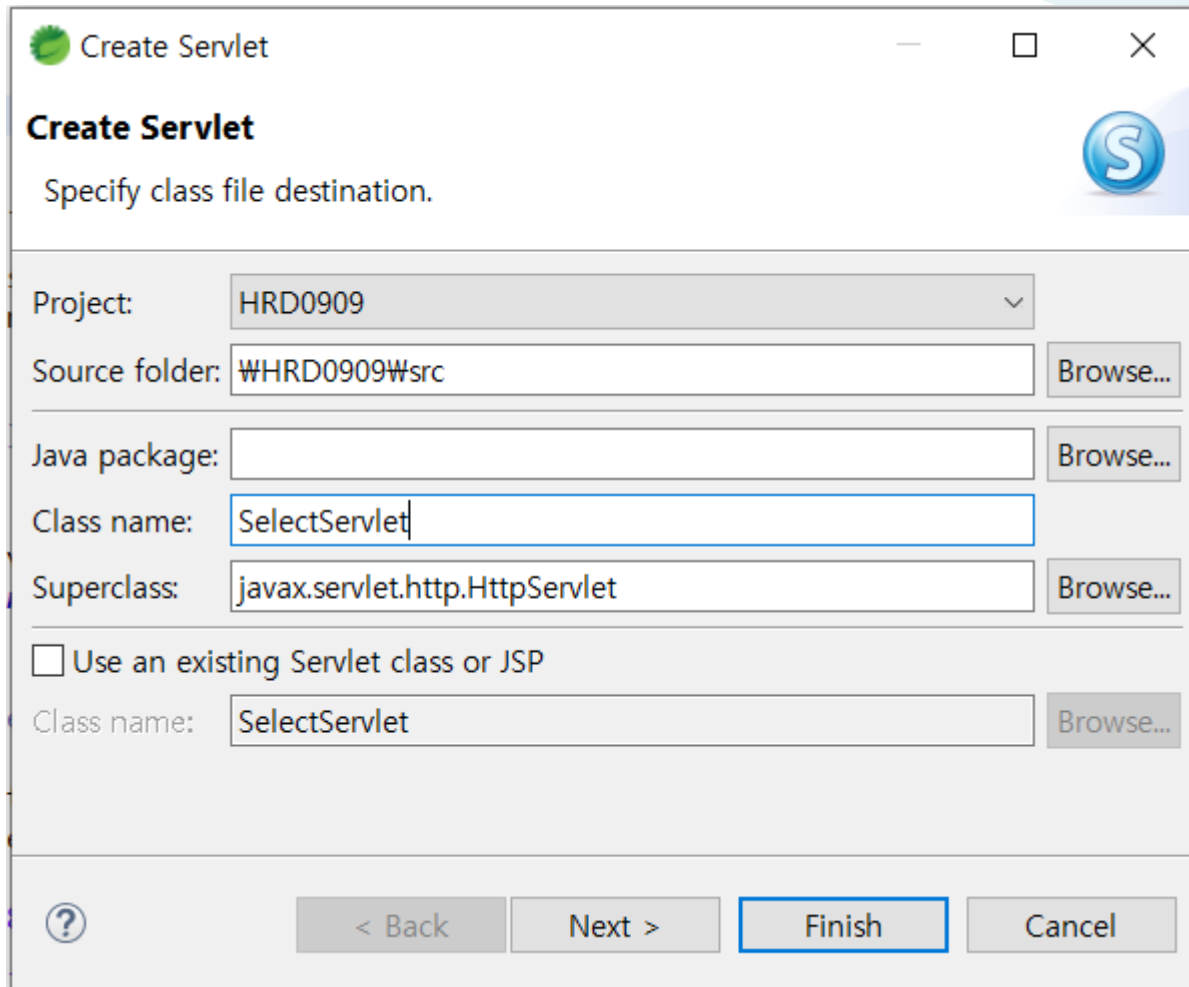
## 2. UserDao 수정

[전체 조회 메서드 추가]

```
30 // 전체 조회
31 public static List<UserVO> selectAllUsers() {
32     ArrayList<UserVO> list = new ArrayList<UserVO>();
33     // Step 1: Establishing a Connection
34     String sql = "select * from userTbl";
35     try (
36         Connection conn = getConnection();
37         PreparedStatement pstmt = conn.prepareStatement(sql);) {
38         ResultSet rs = pstmt.executeQuery();
39
40         while (rs.next()) {
41             int no=rs.getInt("no");
42             String name = rs.getString("name");
43             String email = rs.getString("email");
44             String pwd = rs.getString("pwd");
45             UserVO vo = new UserVO(no,name,email,pwd);
46             list.add(vo);
47         }
48     } catch (SQLException e) {
49         e.printStackTrace();
50     }
51     return list;
52 }
53
```

# 조회하기

## 3. SelectServlet 추가



The image shows a 'Create Servlet' dialog box in an IDE. The title bar says 'Create Servlet'. Below the title bar, there's a 'Create Servlet' section with a blue 'S' icon and the text 'Specify class file destination.'.

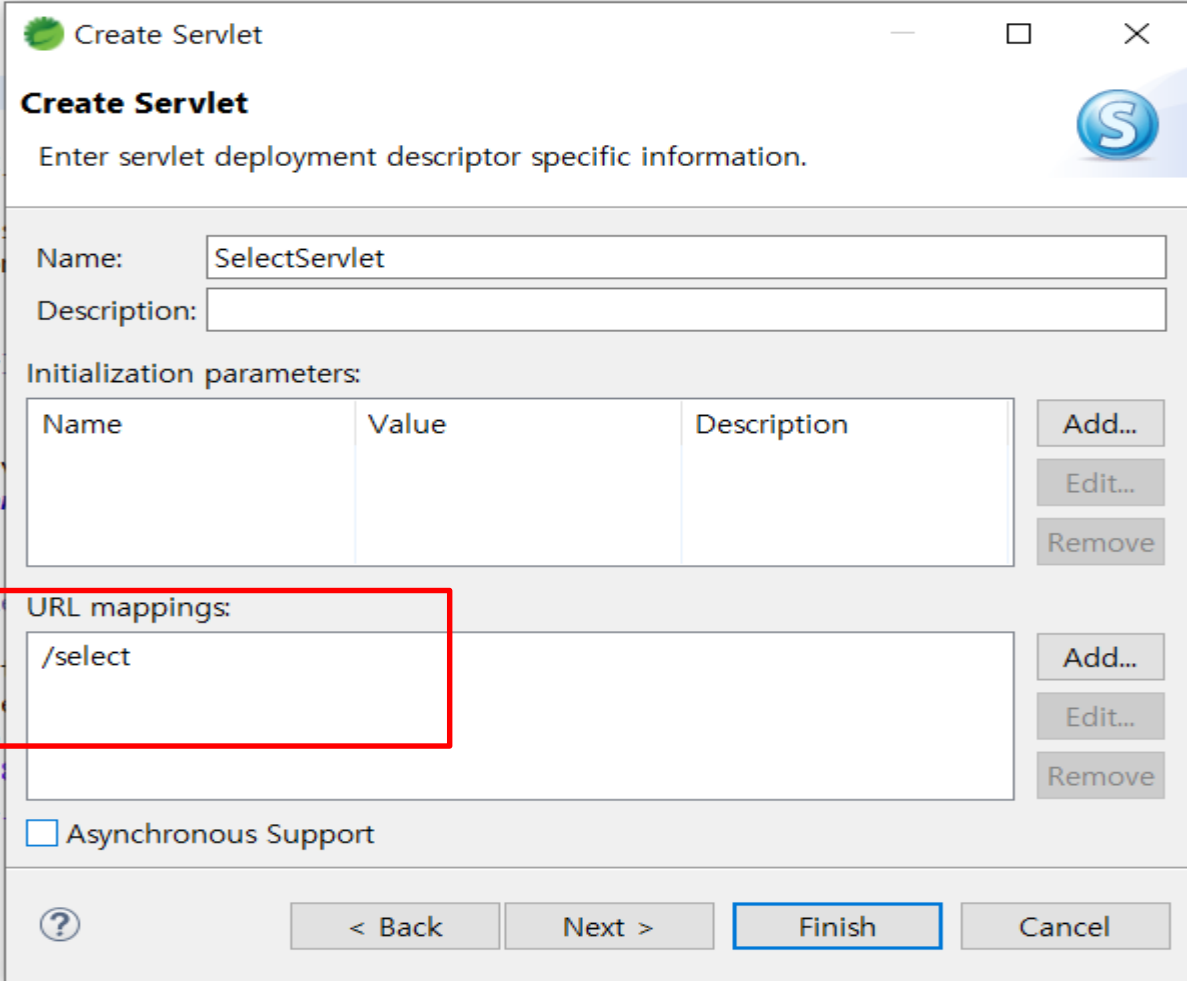
The dialog contains several input fields and buttons:

- Project:** A dropdown menu showing 'HRD0909'.
- Source folder:** A text field containing 'WHRD0909Wsrc' and a 'Browse...' button.
- Java package:** An empty text field and a 'Browse...' button.
- Class name:** A text field containing 'SelectServlet'.
- Superclass:** A text field containing 'javax.servlet.http.HttpServlet' and a 'Browse...' button.
- ☐ **Use an existing Servlet class or JSP**
- Class name:** A text field containing 'SelectServlet' and a 'Browse...' button.

At the bottom, there are four buttons: a help icon (?), '< Back', 'Next >', and 'Finish' (which is highlighted with a blue border). A 'Cancel' button is also present.

# 조회하기

## 3. SelectServlet 추가



The image shows a 'Create Servlet' dialog box from an IDE. The title bar says 'Create Servlet'. Inside, the title is 'Create Servlet' and the instruction is 'Enter servlet deployment descriptor specific information.' There is a blue 'S' icon in the top right. The 'Name' field contains 'SelectServlet'. The 'Description' field is empty. Below is a table for 'Initialization parameters' with columns 'Name', 'Value', and 'Description', and buttons 'Add...', 'Edit...', and 'Remove'. The 'URL mappings' section has a text area containing '/select', with buttons 'Add...', 'Edit...', and 'Remove' to its right. A checkbox for 'Asynchronous Support' is at the bottom left. At the bottom are buttons for '?', '< Back', 'Next >', 'Finish' (highlighted with a blue border), and 'Cancel'.

**Create Servlet**

Enter servlet deployment descriptor specific information.

Name:

Description:

Initialization parameters:

Name	Value	Description
------	-------	-------------

Add... Edit... Remove

URL mappings:

Add... Edit... Remove

☐ Asynchronous Support

? < Back Next > Finish Cancel

# 조회하기

## 3. SelectServlet 추가

```
3 import java.io.IOException;
4 import java.io.PrintWriter;
5 import java.sql.SQLException;
6 import java.util.List;
7 import javax.servlet.ServletException;
8 import javax.servlet.annotation.WebServlet;
9 import javax.servlet.http.HttpServlet;
10 import javax.servlet.http.HttpServletRequest;
11 import javax.servlet.http.HttpServletResponse;
12 /**
13  * Servlet implementation class SelectServlet
14  */
15 @WebServlet("/select")
16 public class SelectServlet extends HttpServlet {
17     private static final long serialVersionUID = 1L;
18     /**
19      * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse response)
20      */
21     protected void doGet(HttpServletRequest request, HttpServletResponse response)
22         throws ServletException, IOException {
23         // TODO Auto-generated method stub
24         request.setCharacterEncoding("utf-8");
25         response.setContentType("text/html;charset=utf-8");
26         try {
27             listUser(request, response);
28         } catch (SQLException ex) {
29             throw new ServletException(ex);
30         }
31     }
```

# 조회하기

## 3. SelectServlet 추가

```
32 private void listUser(HttpServletRequest request, HttpServletResponse response)
33     throws SQLException, IOException, ServletException {
34     List < UserVO > listUser = UserDAO.selectAllUsers();
35     request.setAttribute("listUser", listUser);
36     PrintWriter out = response.getWriter();
37     out.print("<h2>사용자 리스트 </h2><br>");
38     out.print("<html><head><title>hhh</title></head><body>");
39     out.print("<table border=1 width=60%>");
40     out.print("<tr><td> no </td><td> name </td><td> email </td><td> pwd </td></tr><br>");
41     for(int i=0;i<listUser.size();i++) {
42         UserVO userVO =(UserVO) listUser.get(i);
43         int no = userVO.getNo();
44         String name = userVO.getName();
45
46         out.print("<tr><td>"+userVO.getNo()+"</td>");
47         out.print("<td>"+userVO.getName()+"</td>");
48         out.print("<td>"+userVO.getEmail()+"</td>");
49         out.print("<td>"+userVO.getPwd()+"</td></tr><br>");
50     }
51     out.print("</table></body></html>");
52 }
53
```

# 조회하기

## 3. SelectServlet 추가

```
57     protected void doPost(HttpServletRequest request, HttpServletResponse response)
58         throws ServletException, IOException {
59         // TODO Auto-generated method stub
60         doGet(request, response);
61     }
62
63 }
```




# 조회하기

#### 4. 조회 form추가

```
1 <%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>  
2 <!DOCTYPE html>  
3 <html>  
4 <head>  
5     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">  
6     <title>Insert title here</title>  
7 </head>  
8 <body>  
9     <center>  
10    <h1>사용자 관리</h1>  
11    <h2>  
12        &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~  
13    <a href="select">사용자 리스트</a>  
14    </h2>  
15 </center>  
16  
17 </body>
```

# 조회하기

## 5. 실행



사용자 관리

사용자 리스트

# 조회하기

## 5. 실행

### 사용자 리스트

no	name	email	pwd
1234	홍길동	k001	1234
3421	홍길동2	k001	1111

# 실습하기

- ❖ 회원정보 목록(서블릿으로 구현)
- ✓ 회원정보를 조회한다.

← → 🏠 🔍

http://localhost:8080/hrd1007/select\_member.jsp

▶

(과정평가형 정보처리산업기사)웹 쇼핑몰 관리 시스템ver 1.0

회원 정보 등록

상품정보 등록

회원 정보 목록

상품정보 목록

홈으로

회원 정보 조회

id	성명	패스워드	성별	생년월일	이메일	휴대폰	주 소	닉네임
2019100501	홍길동	1234	남성	1990-10-21 00:00:00	hong@kor.com	010-1111-1111	부산시 부산진구 양정동	길동이
2019100502	홍길순	1234	여성	1993-02-23 00:00:00	hongs@kor.com	010-2222-1111	부산시 동래구 수안동	길순이
2019100503	이순신	1234	남성	1991-02-12 00:00:00	lee@kor.com	010-2222-3331	부산시 금정구 장전동	순신이
2019100504	강감찬	1234	남성	1989-02-24 00:00:00	kang@kor.com	010-4444-3331	부산시 해운대구 반송동	찬이
2019100505	이순심	1234	여성	1992-11-24 00:00:00	lees@kor.com	010-4444-5661	부산시 영도구 봉래동	심이
2019100506	홍의장	1234	0	1990-01-01 00:00:00	hong01@naver.com	010-1111-2111	부산시 부산진구 양정동	홍의장군

HRDKOREA Copyright©2019 All rights reserved. Human Resources Development Service of Korea.

- ❖ 상품정보 목록(select\_product.jsp)
- ✓ 상품정보를 조회한다.

### ❖ 상품정보 목록(select\_product.jsp)

✓ 상품정보를 조회한다.

