



롯데 6차 JAVA 백엔드

교과목: 서버구현(JSP)

훈련생: 손준오

평가일시: 2025-03-28

목차

User1 entity 생성

작업 1. **User1** 테이블 데이터 저장

작업 2. **User1** 테이블 데이터 조회

작업 3. **User1** 테이블 데이터 수정

작업 4. **User1** 테이블 데이터 삭제

User1 entity 생성

User1.java (entity) 생성

DB의 데이터를 자바 객체로 변환하여 쉽게 관리하기 위해 사용

```
1 package entity;
2
3 public class User1 {
4
5     private String uid;
6     private String name;
7     private String hp;
8     private int age;
9     public String getUid() {
10         return uid;
11     }
12     public void setUid(String uid) {
13         this.uid = uid;
14     }
15     public String getName() {
16         return name;
17     }
18     public void setName(String name) {
19         this.name = name;
20     }
21     public String getHp() {
22         return hp;
23     }
24     public void setHp(String hp) {
25         this.hp = hp;
26     }
27     public int getAge() {
28         return age;
29     }
30     public void setAge(int age) {
31         this.age = age;
32     }
33
34     @Override
35     public String toString() {
36         return "User1 [uid=" + uid + ", name=" + name + ", hp=" + hp + ", age=" + age + "]";
37     }
38
39
40
41 }
42
```

작업 1. User1 테이블 데이터 저장

← → ↻ ⓘ localhost:8080/ch06/user1/register.jsp

user1 등록

[처음으로 목록이동](#)

아이디	아이디 입력
이름	이름 입력
휴대폰	휴대폰입력
나이	나이 입력
<input type="button" value="등록하기"/>	



← → ↻ ⓘ localhost:8080/ch06/user1/register.jsp

user1 등록

[처음으로 목록이동](#)

아이디	a101
이름	김유신
휴대폰	010-1234-0001
나이	21
<input type="button" value="등록하기"/>	



← → ↻ ⓘ localhost:8080/ch06/user1/list.jsp

user1 목록

[처음으로 등록하기](#)

아이디	이름	휴대폰	나이	관리
a101	김유신	010-1234-0001	21	수정 삭제

User1에 김유신 정보를 등록 했을 때 목록창에서 확인 가능

작업1. User1 테이블 데이터 저장 - 코드 리뷰

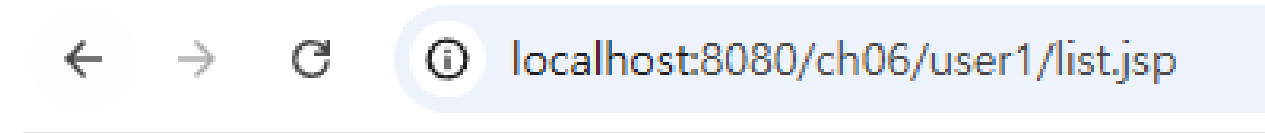
```
1 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
2 <!DOCTYPE html>
3 <html>
4 <head>
5   <meta charset="UTF-8">
6   <title>user1::register</title>
7 </head>
8 <body>
9   <h3>user1 등록</h3>
10  <a href="../1.jdbc.jsp">처음오르</a>
11  <a href="../list.jsp">목록이동</a>
12
13  <form action="../proc/register.jsp" method="post">
14    <table border="1">
15      <tr>
16        <td>아이디</td>
17        <td><input type="text" name="uid" placeholder="아이디 입력"></td>
18      </tr>
19      <tr>
20        <td>이름</td>
21        <td><input type="text" name="name" placeholder="이름 입력"></td>
22      </tr>
23      <tr>
24        <td>휴대폰</td>
25        <td><input type="text" name="hp" placeholder="휴대폰입력"></td>
26      </tr>
27      <tr>
28        <td>나이</td>
29        <td><input type="number" name="age" placeholder="나이 입력"></td>
30      </tr>
31      <tr>
32        <td colspan="2" align="right">
33          <input type="submit" value="등록하기">
34        </td>
35      </tr>
36    </table>
37  </form>
38 </body>
39 </html>
```

User1/register.jsp 코드
> View 페이지 구현

```
3 <%@page import="javax.naming.Context"%>
4 <%@page import="java.sql.PreparedStatement"%>
5 <%@page import="java.sql.Connection"%>
6 <%@page import="java.net.ConnectException"%>
7 <%@page import="java.sql.DriverManager"%>
8 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
9
10 <%
11 // 클라이언트로부터 전달된 데이터 가져오기
12 String uid = request.getParameter("uid"); // 사용자 아이디
13 String name = request.getParameter("name"); // 사용자 이름
14 String hp = request.getParameter("hp"); // 휴대폰 번호
15 String age = request.getParameter("age"); // 나이 (문자열로 받아옴)
16
17 /*
18 // 데이터베이스 접속 정보 (사용하지 않는 코드 - JNDI 사용)
19 String host = "jdbc:mysql://127.0.0.1:3306/studydb";
20 String user = "root";
21 String pass = "abc1234";
22 */
23
24 try {
25   // 1. JNDI 서비스 객체 생성
26   Context initCtx = new InitialContext();
27   Context ctx = (Context) initCtx.lookup("java:comp/env"); // JNDI 기본 환경 이름
28
29   // 2. 커넥션 풀에서 커넥션 가져오기
30   DataSource ds = (DataSource) ctx.lookup("jdbc/studydb");
31   Connection conn = ds.getConnection();
32
33   // 3. SQL 실행을 위한 PreparedStatement 객체 생성
34   String sql = "INSERT INTO `user1` VALUES(?,?,?,?)";
35   PreparedStatement psmt = conn.prepareStatement(sql);
36   psmt.setString(1, uid); // 첫 번째 ? 에 uid 값 설정
37   psmt.setString(2, name); // 두 번째 ? 에 name 값 설정
38   psmt.setString(3, hp); // 세 번째 ? 에 hp 값 설정
39   psmt.setString(4, age); // 네 번째 ? 에 age 값 설정 (주의: 나이는 int 변환 필요할 수도 있음)
40
41   // 4. SQL 실행 (데이터 삽입)
42   psmt.executeUpdate();
43
44   // 5. 사용한 리소스 정리 (데이터베이스 연결 종료)
45   psmt.close();
46   conn.close();
47
48 } catch (Exception e) {
49   e.printStackTrace(); // 예외 발생 시 콘솔에 출력
50 }
51
52 // 데이터 삽입 후 목록 페이지로 이동
53 response.sendRedirect("../list.jsp");
54 %>
```

User1//proc/register.jsp 코드
> Register 기능 구현

작업 2. User1 테이블 데이터 조회



user1 목록

[처음으로 등록하기](#)

아이디	이름	휴대폰	나이	관리
a101	김유신	010-1234-0001	21	수정 삭제
a102	김춘추	010-1234-0002	20	수정 삭제
a103	장보고	010-1234-0003	31	수정 삭제

User1 테이블 데이터 **list**를 조회 할 수 있다.

작업 2. User1 테이블 데이터 조회 - 코드 리뷰

```
1 <%@page import="javax.sql.DataSource"%>
2 <%@page import="javax.naming.Context"%>
3 <%@page import="javax.naming.InitialContext"%>
4 <%@page import="java.sql.ResultSet"%>
5 <%@page import="entity.User1"%> <!-- User1 엔티티 클래스 임포트 -->
6 <%@page import="java.util.ArrayList"%>
7 <%@page import="java.util.List"%>
8 <%@page import="java.sql.DriverManager"%>
9 <%@page import="java.sql.Connection"%>
10 <%@page import="java.sql.Statement"%>
11 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
12
13 <%
14 // 사용자 목록을 저장할 리스트 생성
15 List<User1> users = new ArrayList<>();
16
17 try {
18 // JNDI 서비스 객체 생성 (데이터베이스 연결을 위해)
19 Context initCtx = new InitialContext();
20 Context ctx = (Context) initCtx.lookup("java:comp/env"); // JNDI 기본 환경 검색
21
22 // 커넥션 풀에서 커넥션 가져오기
23 DataSource ds = (DataSource) ctx.lookup("jdbc/studydb");
24 Connection conn = ds.getConnection();
25
26 // SQL 실행을 위한 Statement 객체 생성
27 Statement stmt = conn.createStatement();
28
29 // user1 테이블에서 모든 데이터 조회
30 ResultSet rs = stmt.executeQuery("SELECT * FROM `user1`");
31
32 // 조회된 결과를 User1 객체에 저장하고 리스트에 추가
33 while(rs.next()) {
34     User1 user1 = new User1();
35     user1.setUid(rs.getString(1)); // 첫 번째 컬럼: uid
36     user1.setName(rs.getString(2)); // 두 번째 컬럼: name
37     user1.setHp(rs.getString(3)); // 세 번째 컬럼: hp
38     user1.setAge(rs.getInt(4)); // 네 번째 컬럼: age
39     users.add(user1); // 리스트에 추가
40 }
41
42 // 자원 해제 (메모리 누수 방지)
43 rs.close();
44 stmt.close();
45 conn.close();
46
47 } catch (Exception e) {
48     e.printStackTrace(); // 예외 발생 시 콘솔에 오류 출력
49 }
50 %>
```

```
49 }
50 %>
51
52 <!DOCTYPE html>
53 <html>
54 <head>
55     <meta charset="UTF-8"%>
56     <title>user1::list</title>
57 </head>
58 <body>
59     <h3>user1 목록</h3>
60     <!-- 다른 페이지로 이동하는 링크 -->
61     <a href="/1.jdbc.jsp">처음으로</a>
62     <a href="/register.jsp">등록하기</a>
63
64     <!-- 사용자 목록을 출력하는 테이블 -->
65     <table border="1">
66         <tr>
67             <th>아이디</th>
68             <th>이름</th>
69             <th>휴대폰</th>
70             <th>나이</th>
71             <th>관리</th>
72         </tr>
73
74         <!-- users 리스트를 반복하여 테이블에 데이터 출력 -->
75         <% for(User1 user1 : users) { %>
76         <tr>
77             <td><%= user1.getUid() %></td> <!-- uid 출력 -->
78             <td><%= user1.getName() %></td> <!-- name 출력 -->
79             <td><%= user1.getHp() %></td> <!-- hp 출력 -->
80             <td><%= user1.getAge() %></td> <!-- age 출력 -->
81             <td>
82                 <!-- 수정 버튼 -->
83                 <a href="/modify.jsp?uid=<%= user1.getUid() %>">수정</a>
84                 <!-- 삭제 버튼 (JavaScript로 삭제 확인 창 출력) -->
85                 <a href="javascript:void(0);" onclick="confirmDelete('<%= user1.getUid() %>')">삭제</a>
86             </td>
87         </tr>
88         <% } %> <!-- for 문 종료 -->
89     </table>
90 </body>
91 </html>
92
```

User1/list.jsp 코드 리뷰(주석 참조)

작업 3. User1 테이블 데이터 수정

localhost:8080/ch06/user1/list.jsp

user1 목록

[처음으로 등록하기](#)

아이디	이름	휴대폰	나이	관리
a101	김유신	010-1234-0001	21	수정 삭제
a102	김춘추	010-1234-0002	20	수정 삭제
a103	장보고	010-1234-0003	31	수정 삭제



localhost:8080/ch06/user1/modify.jsp?uid=a101

user1 수정

[처음으로 목록이동](#)

아이디	<input type="text" value="a101"/>
이름	<input type="text" value="김유신"/>
휴대폰	<input type="text" value="010-1234-0001"/>
나이	<input type="text" value="21"/>
<input type="button" value="수정하기"/>	



localhost:8080/ch06/user1/modify.jsp?uid=a101

user1 수정

[처음으로 목록이동](#)

아이디	<input type="text" value="a101"/>
이름	<input type="text" value="김유신투"/>
휴대폰	<input type="text" value="010-1234-0008"/>
나이	<input type="text" value="39"/>
<input type="button" value="수정하기"/>	



A101 김 유 신 데이터를 수정
눌렀을때, 수정 페이지에
김 유 신 데이터가 출력 된다.
(uid101 db 값을 가져옴

> uid 값은 수정이
불가능하다(read only)

localhost:8080/ch06/user1/list.jsp

user1 목록

[처음으로 등록하기](#)

아이디	이름	휴대폰	나이	관리
a101	김유신투	010-1234-0000	39	수정 삭제
a102	김춘추	010-1234-0002	20	수정 삭제
a103	장보고	010-1234-0003	31	수정 삭제

작업 3. User1 테이블 데이터 수정 - 코드 리뷰

```
1 <%@page import="javax.sql.DataSource"%>
2 <%@page import="javax.naming.InitialContext"%>
3 <%@page import="javax.naming.Context"%>
4 <%@page import="entity.User1"%>
5 <%@page import="java.sql.ResultSet"%>
6 <%@page import="java.sql.PreparedStatement"%>
7 <%@page import="java.sql.Connection"%>
8 <%@page import="java.sql.DriverManager"%>
9 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
10 <%
11     String uid = request.getParameter("uid");
12
13     //데이터베이스 처리
14     //String host = "jdbc:mysql://127.0.0.1:3306/studydb";
15     //String user = "root";
16     //String pass = "abc1234";
17
18     User1 user1 = null; // 선언
19
20     try {
21         //Class.forName("com.mysql.cj.jdbc.Driver");
22         //Connection conn = DriverManager.getConnection(host, user, pass);
23
24         //dbcp
25         Context ctx =(Context) new InitialContext().lookup("java:comp/env");
26         DataSource ds = (DataSource) ctx.lookup("jdbc/studydb");
27         Connection conn =ds. getConnection();
28
29         String sql = "SELECT * FROM `user1` where `uid`=?";
30         PreparedStatement psmt = conn.prepareStatement(sql);
31         psmt.setString(1, uid);
32
33         ResultSet rs = psmt.executeQuery();
34
35         if(rs.next()){
36             user1 = new User1(); // 생성
37             user1.setUid(rs.getString(1));
38             user1.setName(rs.getString(2));
39             user1.setHp(rs.getString(3));
40             user1.setAge(rs.getInt(4));
41         }
42
43         rs.close();
44         psmt.close();
45         conn.close();
46
47     }catch(Exception e){
```

```
50 %>
51 <!DOCTYPE html>
52 <html>
53 <head>
54     <meta charset="UTF-8">
55     <title>user1::modify</title>
56 </head>
57 <body>
58     <h3>user1 수정</h3>
59     <a href=" ../1.jdbc.jsp">처음오르</a>
60     <a href=" ../list.jsp">목록이동</a>
61
62     <form action=" ../proc/modify.jsp" method="post">
63         <table border="1">
64             <tr>
65                 <td>아이디</td>
66                 <td><input type="text" name="uid" value="<%= user1.getUid() %>" readonly></td>
67             </tr>
68             <tr>
69                 <td>이름</td>
70                 <td><input type="text" name="name" value="<%= user1.getName() %>" placeholder="이름 입력"></td>
71             </tr>
72             <tr>
73                 <td>휴대폰</td>
74                 <td><input type="text" name="hp" value="<%= user1.getHp() %>" placeholder="휴대폰 입력"></td>
75             </tr>
76             <tr>
77                 <td>나이</td>
78                 <td><input type="number" name="age" value="<%= user1.getAge() %>" placeholder="나이 입력"></td>
79             </tr>
80             <tr>
81                 <td colspan="2" align="right">
82                     <input type="submit" value="수정하기">
83                 </td>
84             </tr>
85         </table>
86     </form>
87
88 </body>
89 </html>
```

User1/modify.jsp 코드
> View 페이지 구현

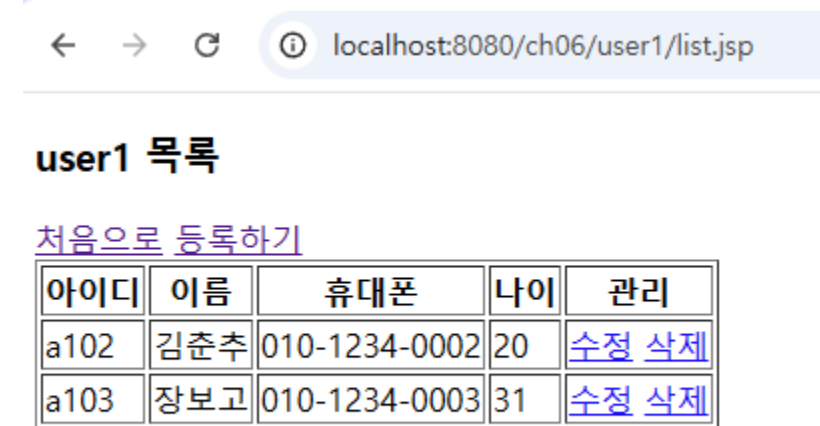
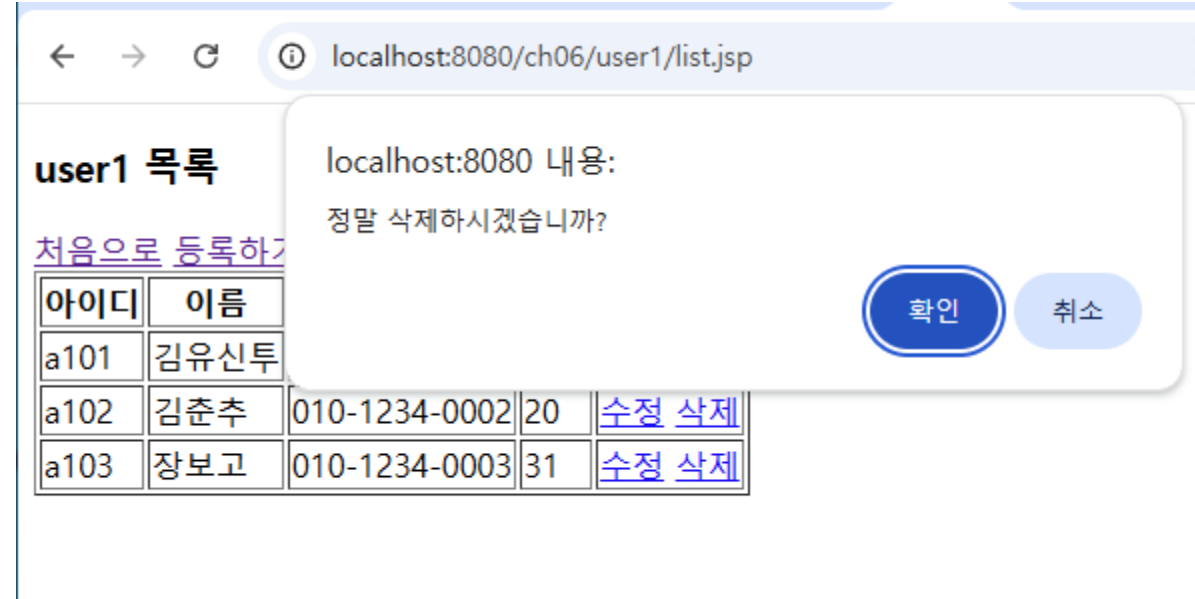
작업 3. User1 테이블 데이터 수정 - 코드 리뷰

```
1 <%@page import="java.sql.PreparedStatement"%>
2 <%@page import="java.sql.DriverManager"%>
3 <%@page import="java.sql.Connection"%>
4 <%@page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
5 <%
6     // 데이터 수신
7     String uid = request.getParameter("uid");
8     String name = request.getParameter("name");
9     String hp = request.getParameter("hp");
10    String age = request.getParameter("age");
11
12    // 데이터베이스 처리
13    String host = "jdbc:mysql://127.0.0.1:3306/studydb";
14    String user = "root";
15    String pass = "1234";
16
17    try {
18        Class.forName("com.mysql.cj.jdbc.Driver");
19        Connection conn = DriverManager.getConnection(host, user, pass);
20
21        String sql = "UPDATE `user1` SET `name`=?, `hp`=?, `age`=? WHERE `uid`=?";
22        PreparedStatement psmt = conn.prepareStatement(sql);
23        psmt.setString(1, name);
24        psmt.setString(2, hp);
25        psmt.setString(3, age);
26        psmt.setString(4, uid);
27
28        psmt.executeUpdate();
29
30        psmt.close();
31        conn.close();
32    } catch (Exception e) {
33        e.printStackTrace();
34    }
35
36    // 목록이동
37    response.sendRedirect("../list.jsp");
38 %>
```

User1/proc/modify.jsp 코드

> 기능 페이지 구현

작업 4. User1 테이블 데이터 삭제



User1 테이블에서 a101 김유탄투 데이터 삭제를 클릭했을 때
여부 확인창 확인 후 확인을 누르면 삭제가 되는것을 확인 가능

작업 4. User1 테이블 데이터 삭제 - 코드 리뷰

```
1 <%@page import="javax.sql.DataSource"%>
2 <%@page import="javax.naming.InitialContext"%>
3 <%@page import="javax.naming.Context"%>
4 <%@page import="java.sql.PreparedStatement"%>
5 <%@page import="java.sql.DriverManager"%>
6 <%@page import="java.sql.Connection"%>
7 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
8 <%
9     // 데이터 수신
10    String uid = request.getParameter("uid");
11
12    // 데이터베이스 처리
13    //String host = "jdbc:mysql://127.0.0.1:3306/studydb";
14    //String user = "root";
15    //String pass = "abc1234";
16
17    try {
18        //Class.forName("com.mysql.cj.jdbc.Driver");
19        //Connection conn = DriverManager.getConnection(host, user, pass);
20
21        //JNDI 서비스 객체 생성
22        Context initCtx = new InitialContext();
23        Context ctx = (Context)initCtx.lookup("java:comp/env"); //JNDI 기본환경 이름
24
25        //커넥션 풀에 있는 커넥션을 가져오기
26        DataSource ds = (DataSource) ctx.lookup("jdbc/studydb");
27        Connection conn = ds.getConnection();
28
29        String sql = "DELETE FROM `user1` WHERE `uid`=?" ;
30        PreparedStatement psmt = conn.prepareStatement(sql);
31        psmt.setString(1, uid);
32
33        psmt.executeUpdate();
34
35        psmt.close();
36        conn.close();
37
38    } catch (Exception e) {
39        e.printStackTrace();
40    }
41
42    // 목록이동
43    response.sendRedirect("../list.jsp");
44 %>
```

```
<td>
    <!-- 수정 버튼 -->
    <a href="/modify.jsp?uid=<%= user1.getId() %>">수정</a>
    <!-- 삭제 버튼 (JavaScript로 삭제 확인 창 출력) -->
    <a href="javascript:void(0);" onclick="confirmDelete('<%= user1.getId() %>')">삭제</a>
</td>
</tr>
```

```
92
93 <!-- JavaScript 코드: 삭제 버튼 클릭 시 확인 창 띄우기 -->
94 <script>
95     function confirmDelete(uid) {
96         if (confirm("정말 삭제하시겠습니까?")) {
97             location.href = "/proc/delete.jsp?uid=" + uid; // 확인 클릭 시 삭제 요청 전송
98         }
99     }
100 </script>
```

**User1/list.jsp 코드에
삭제 버튼 코드 수정 및
Script 문구 추가**

**User1/proc/delete.jsp 코드
> 기능 페이지 구현**



감사합니다.