
05. SQL 활용 수행평가

JSP로 SQL 활용하기

날짜 : 2022/10/17

이름 : 조주영



INDEX

1

프로젝트 생성 및 구성

2

화면 구현

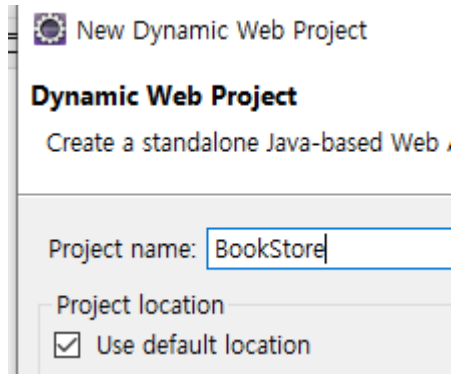
3

기능 구현

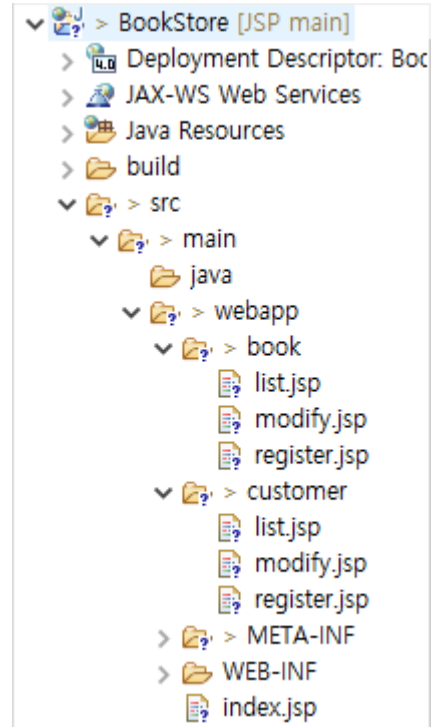
4

실행

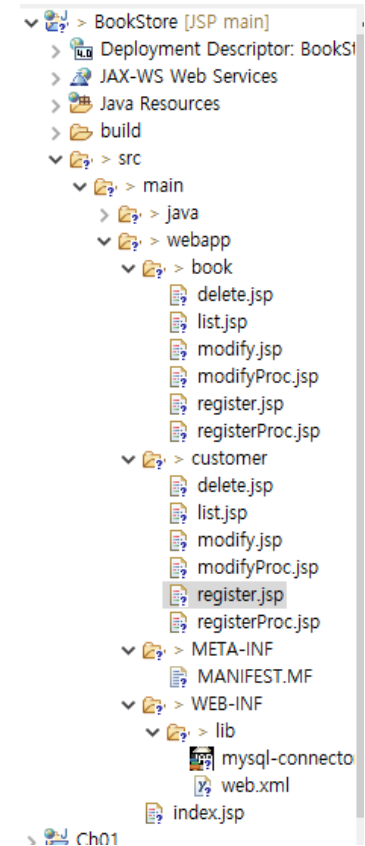
1. 프로젝트 생성 및 구성



Dynamic Web Project 로
BookStore 생성



초기 구성



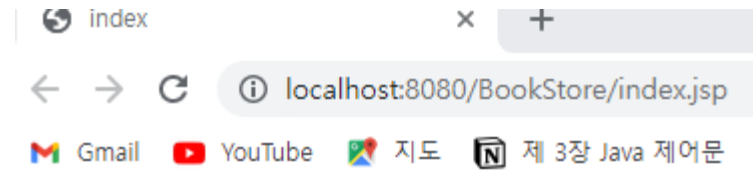
최종 구성

1. 프로젝트 생성 및 구성

```
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>index</title>
7     <!--
8       날짜 : 2022/10/17
9       이름 : 조주영
10      내용 : 5회차 수행평가 SQL 활용
11    -->
12   </head>
13   <body>
14     <h3>Bookstore</h3>
15
16     <a href="./book/list.jsp">도서목록</a>
17     <a href="./customer/list.jsp">고객목록</a>
18
19   </body>
20 </html>
```

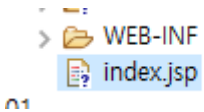


Run as



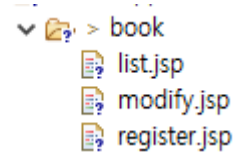
Bookstore

[도서목록](#) [고객목록](#)

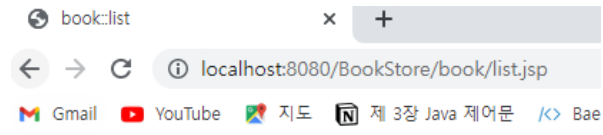


Index.jsp

2. 화면 구현



Book 폴더 구현 화면

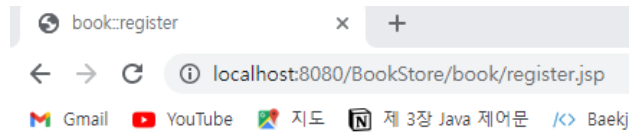


도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	죽구의 역사	굿스포츠	7000	수정 삭제

list.jsp

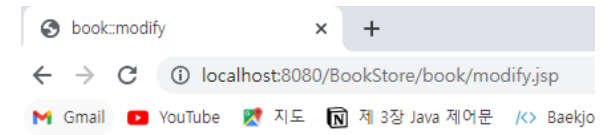


도서등록

[처음으로 도서목록](#)

도서번호	<input type="text"/>
도서명	<input type="text"/>
출판사	<input type="text"/>
가격	<input type="text"/>
<input type="button" value="등록"/>	

register.jsp



도서수정

[처음으로 도서목록](#)

도서번호	<input type="text"/>
도서명	<input type="text"/>
출판사	<input type="text"/>
가격	<input type="text"/>
<input type="button" value="수정"/>	

modify.jsp

2. 화면 구현

customer
list.jsp
modify.jsp
register.jsp

Customer 폴더 구현 화면



customer::list

localhost:8080/BookStore/customer/list.jsp

고객목록

[처음으로](#) [고객등록](#)

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스터	010-1234-1001	수정 삭제

list.jsp

customer::register

localhost:8080/BookStore/customer/register.jsp

고객등록

[처음으로](#) [고객목록](#)

고객번호	<input type="text"/>
고객명	<input type="text"/>
주소	<input type="text"/>
휴대폰	<input type="text"/>
<input type="button" value="등록"/>	

register.jsp

customer::modify

localhost:8080/BookStore/customer/modify.jsp

고객수정

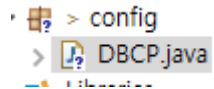
[처음으로](#) [고객목록](#)

고객번호	<input type="text"/>
고객명	<input type="text"/>
주소	<input type="text"/>
휴대폰	<input type="text"/>
<input type="button" value="수정"/>	

modify.jsp

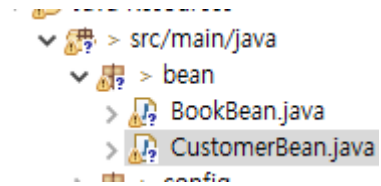
3. 기능 구현

DBCP 생성



Code

Bean 생성



```
1 package config;
2
3 import java.sql.Connection;
4 import java.sql.SQLException;
5
6 import javax.naming.InitialContext;
7 import javax.naming.NamingException;
8 import javax.sql.DataSource;
9
10 public class DBCP {
11
12     private static DataSource ds = null;
13
14     public static Connection getConnection() throws NamingException, SQLException {
15
16         if (ds == null) {
17             ds = (DataSource) new InitialContext()
18                 .lookup("java:comp/env/dbcp_java2_bookstore");
19         }
20
21         return ds.getConnection();
22     }
23 }
24
25 }
26
```

3. 기능 구현

book/list

페이지 관계로 코드 첨부는 book 폴더 코드들만 했습니다.
똑같은 코드로 bean과 column명 등만 약간 수정하였습니다.

```
1 <%@page import="java.util.ArrayList"%>
2 <%@page import="java.sql.ResultSet"%>
3 <%@page import="java.sql.Statement"%>
4 <%@page import="config.DBCP"%>
5 <%@page import="java.sql.Connection"%>
6 <%@page import="bean.BookBean"%>
7 <%@page import="java.util.List"%>
8 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
9 <%
10
11     List<BookBean> books = null;
12
13     try {
14         Connection conn = DBCP.getConnection();
15         Statement stmt = conn.createStatement();
16         ResultSet rs = stmt.executeQuery("SELECT * FROM `book`");
17         books = new ArrayList<>();
18
19         while(rs.next()){
20
21             BookBean bb = new BookBean();
22             bb.setBookId(rs.getInt(1));
23             bb.setBookName(rs.getString(2));
24             bb.setPublisher(rs.getString(3));
25             bb.setPrice(rs.getInt(4));
26
27             books.add(bb);
28
29         }
30         rs.close();
31         stmt.close();
32         conn.close();
33     } catch (Exception e){
34         e.printStackTrace();
35     }
36
37 %>
```

```
38 <!DOCTYPE html>
39 <html>
40     <head>
41         <meta charset="UTF-8">
42         <title>book::list</title>
43     </head>
44     <body>
45         <h3>도서목록</h3>
46
47         <a href=" ../index.jsp">처음으로</a>
48         <a href=" ../register.jsp">도서등록</a>
49
50         <table border="1">
51             <tr>
52                 <th>도서번호</th>
53                 <th>도서명</th>
54                 <th>출판사</th>
55                 <th>가격</th>
56                 <th>관리</th>
57             </tr>
58             <% for(BookBean bb : books){ %>
59             <tr>
60                 <td><%= bb.getBookId() %></td>
61                 <td><%= bb.getBookName() %></td>
62                 <td><%= bb.getPublisher() %></td>
63                 <td><%= bb.getPrice() %></td>
64                 <th><a href=" ../modify.jsp?bookId=<%= bb.getBookId() %>">수정</a>
65                     <a href=" ../delete.jsp?bookId=<%= bb.getBookId() %>">삭제</a></th>
66             </tr>
67             <% } %>
68         </table>
69
70     </body>
71 </html>
```


3. 기능 구현

book/register

```
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>book::register</title>
7 </head>
8 <body>
9 <h3>도서등록</h3>
10
11 <a href="../index.jsp">처음으로</a>
12 <a href="/List.jsp">도서목록</a>
13
14 <form action="/registerProc.jsp" method="post">
15 <table border="1">
16 <tr>
17 <td>도서번호</td>
18 <td><input type="text" name="bookId" value="auto" readonly></td>
19 </tr>
20 <tr>
21 <td>도서명</td>
22 <td><input type="text" name="bookName"></td>
23 </tr>
24 <tr>
25 <td>출판사</td>
26 <td><input type="text" name="publisher"></td>
27 </tr>
28 <tr>
29 <td>가격</td>
30 <td><input type="text" name="price"></td>
31 </tr>
32 <tr>
33 <td colspan="2" align="right">
34 <input type="submit" value="등록"/></td>
35 </tr>
36 </table>
37 </form>
38 </body>
39 </html>
```

전송 데이터 수신

registerProc

```
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="config.DBCP"%>
6 <%@page import="java.sql.Connection"%>
7 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
8 <%
9 request.setCharacterEncoding("utf-8");
10
11 String bookId = request.getParameter("bookId");
12 String bookName = request.getParameter("bookName");
13 String publisher = request.getParameter("publisher");
14 String price = request.getParameter("price");
15
16 try {
17 Context initCtx = new InitialContext();
18 Context ctx = (Context)initCtx.lookup("java:comp/env");
19 DataSource ds = (DataSource)ctx.lookup("dbcp_java2_bookstore");
20 Connection conn = ds.getConnection();
21
22 String sql = "INSERT INTO `book` (`bookName`, `publisher`, `price`) VALUES (?, ?, ?)";
23 PreparedStatement pstmt = conn.prepareStatement(sql);
24 pstmt.setString(1, bookName);
25 pstmt.setString(2, publisher);
26 pstmt.setString(3, price);
27
28 pstmt.executeUpdate();
29
30 pstmt.close();
31 conn.close();
32
33 } catch (Exception e){
34 e.printStackTrace();
35 }
36
37 response.sendRedirect("../list.jsp");
38
39 %>
```

커넥션 풀에서 커넥션 가져오기

JNDI 서비스 객체 생성

커넥션 반납

리다이렉트

3. 기능 구현

book/modify

```
1 <%@page import="java.sql.ResultSet"%>
2 <%@page import="java.sql.PreparedStatement"%>
3 <%@page import="config.DBCP"%>
4 <%@page import="java.sql.Connection"%>
5 <%@page import="bean.BookBean"%>
6 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
7 <%
8
9 request.setCharacterEncoding("utf-8");
10 String bookId = request.getParameter("bookId");
11
12 BookBean bb = new BookBean();
13
14 try{
15     Connection conn = DBCP.getConnection();
16     String sql = "SELECT * FROM `book` WHERE `bookId`=?";
17     PreparedStatement psmt = conn.prepareStatement(sql);
18     psmt.setString(1, bookId);
19
20     ResultSet rs = psmt.executeQuery();
21
22     if (rs.next()){
23         bb.setBookId(rs.getInt(1));
24         bb.setBookName(rs.getString(2));
25         bb.setPublisher(rs.getString(3));
26         bb.setPrice(rs.getInt(4));
27     }
28
29     rs.close();
30     psmt.close();
31     conn.close();
32 } catch (Exception e){
33     e.printStackTrace();
34 }
35 }
36
37 %>
```

```
38 <!DOCTYPE html>
39 <html>
40 <head>
41     <meta charset="UTF-8">
42     <title>book::modify</title>
43 </head>
44 <body>
45     <h3>도서수정</h3>
46
47     <a href=".."index.jsp">처음으로</a>
48     <a href=".."List.jsp">도서목록</a>
49
50 <form action=".."modifyProc.jsp" method="post">
51     <table border="1">
52     <tr>
53         <td>도서번호</td>
54         <td><input type="text" name="bookId" readonly value="<%= bb.getBookId() %>"></td>
55     </tr>
56 <tr>
57         <td>도서명</td>
58         <td><input type="text" name="bookName" placeholder="<%= bb.getBookName() %>"></td>
59     </tr>
60 <tr>
61         <td>출판사</td>
62         <td><input type="text" name="publisher" placeholder="<%= bb.getPublisher() %>"></td>
63     </tr>
64 <tr>
65         <td>가격</td>
66         <td><input type="text" name="price" placeholder="<%= bb.getPrice() %>"></td>
67     </tr>
68 <tr>
69         <td colspan="2" align="right">
70             <input type="submit" value="수정">
71         </td>
72     </tr>
73 </table>
74 </form>
75 </body>
76 </html>
```

전송 데이터 수신

데이터베이스 작업

modifyProc

```
1 <%@page import="java.sql.PreparedStatement"%>
2 <%@page import="config.DBCP"%>
3 <%@page import="java.sql.Connection"%>
4 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
5 <%
6
7 request.setCharacterEncoding("utf-8");
8
9 String bookId = request.getParameter("bookId");
10 String bookName = request.getParameter("bookName");
11 String publisher = request.getParameter("publisher");
12 String price = request.getParameter("price");
13
14 try {
15     Connection conn = DBCP.getConnection();
16     String sql = "UPDATE `book` SET `bookName`=?, `publisher`=?, `price`=? WHERE `bookId`=?";
17     PreparedStatement psmt = conn.prepareStatement(sql);
18     psmt.setString(1, bookName);
19     psmt.setString(2, publisher);
20     psmt.setString(3, price);
21     psmt.setString(4, bookId);
22
23     psmt.executeUpdate();
24
25     psmt.close();
26     conn.close();
27 } catch (Exception e){
28     e.printStackTrace();
29 }
30
31 response.sendRedirect(".."list.jsp");
32
33 %>
```

커넥션 반납

3. 기능 구현

book/delete

```
1 <%@page import="java.sql.PreparedStatement"%>
2 <%@page import="config.DBCP"%>
3 <%@page import="java.sql.Connection"%>
4 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
5 <%
6     request.setCharacterEncoding("utf-8");
7     String bookId = request.getParameter("bookId");
8
9     try {
10         Connection conn = DBCP.getConnection();
11         String sql = "DELETE FROM `book` WHERE `bookId`=?" ;
12         PreparedStatement psmt = conn.prepareStatement(sql);
13         psmt.setString(1, bookId);
14
15         psmt.executeUpdate();
16         psmt.close();
17         conn.close();
18
19     } catch (Exception e){
20         e.printStackTrace();
21     }
22
23     response.sendRedirect("../list.jsp");
24
25 %>
```

4. 실행

book

book:register

localhost:8080/BookStore/

도서등록

처음으로 도서목록

도서번호	auto
도서명	문학의 이해
출판사	동아대학교
가격	15000
등록	

book:list

localhost:8080/BookStore/book/list.jsp

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champions	Pearson	13000	수정 삭제
11	스포츠의학	한솔의학서적	0	수정 삭제
14	문학의 이해	동아대학교	15000	수정 삭제

수정

book:modify

localhost:8080/BookStore/b

book:modify

localhost:8080/BookStore/bo

도서수정

처음으로 도서목록

도서번호	14
도서명	문학의 이해
출판사	동아대학교
가격	15000
수정	

book:modify

localhost:8080/BookStore/bo

도서수정

처음으로 도서목록

도서번호	14
도서명	문법의 이해
출판사	경성대학교
가격	13000
수정	

book:list

localhost:8080/BookStore/book/list.jsp

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champions	Pearson	13000	수정 삭제
11	스포츠의학	한솔의학서적	0	수정 삭제
14	문법의 이해	경성대학교	13000	수정 삭제

삭제

book:list

localhost:8080/BookStore/book/list.jsp

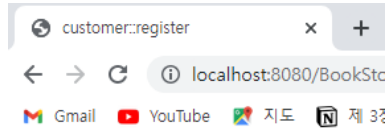
도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	굿스포츠	8000	수정 삭제
6	역도 단계별 기술	굿스포츠	6000	수정 삭제
7	야구의 추억	이상미디어	20000	수정 삭제
8	야구를 부탁해	이상미디어	13000	수정 삭제
9	올림픽 이야기	삼성당	7500	수정 삭제
10	Olympic Champions	Pearson	13000	수정 삭제

4. 실행

customer

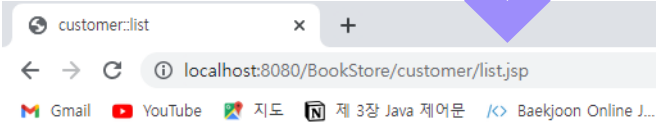


고객등록

처음으로 고객목록

고객번호	auto
고객명	홍길동
주소	울도국
휴대폰	010-1234-1234
<input type="button" value="등록"/>	

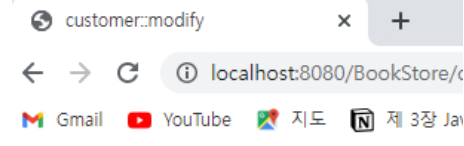
수정



고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
6	박세리	대한민국 대전	null	수정 삭제
9	홍길동	부산시 금정구	010-1234-1234	수정 삭제

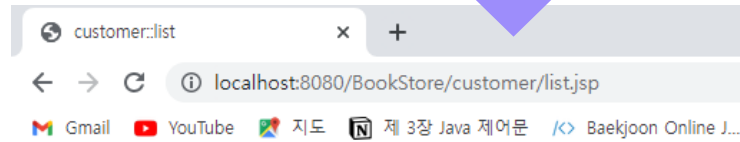


고객수정

처음으로 고객목록

고객번호	9
고객명	홍길동
주소	울도국
휴대폰	010-1234-1234
<input type="button" value="수정"/>	

삭제



고객목록

처음으로 고객등록

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 서울	000-6000-0001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
6	박세리	대한민국 대전	null	수정 삭제
9	홍길동	부산시 금정구	010-1234-1234	수정 삭제