

정기수행평가

SQI 활용

2022/10/21
길은섭

목차

1. 프로젝트 생성 및 구성

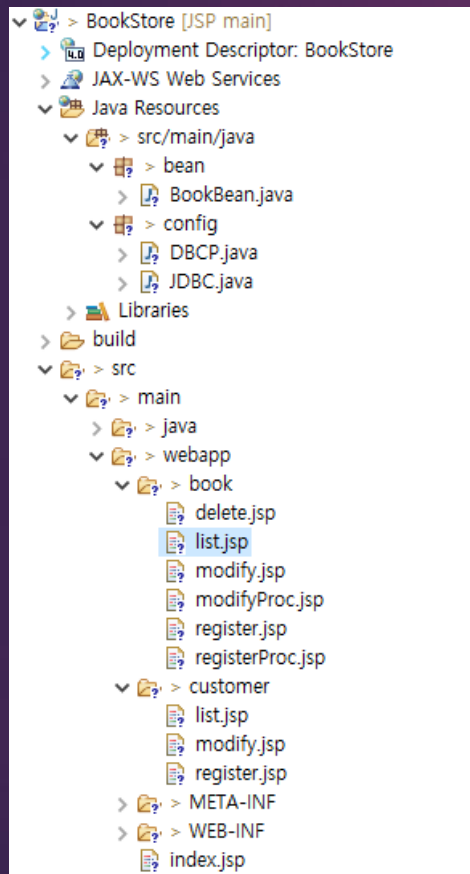
2. 화면 구현

3. 기능 구현

4. 실행

1. 프로젝트 생성 및 구성

프로젝트 구조

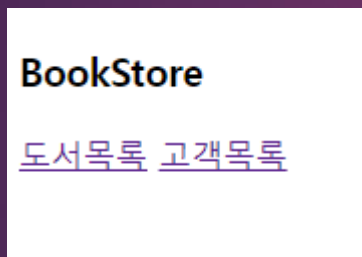


- 프로젝트 명 : BookStore
- WAS: Tomcat 9
- DB : java1_bookstore
- 개발도구 : Eclipse, Workbench
- DBCP로 코드 사용을 편하게 하고 서버 과부하를 줄임
- JavaBean으로 코드의 복잡성을 줄이고 재사용하기 쉽게 만듦

2. 화면 구현

- book(도서)와 customer(고객)의 목록, 등록, 수정 화면 구현을 해보겠습니다.

index



list

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구의 이해	대한미디어	22000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제

register

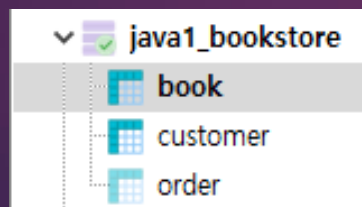
도서등록

[처음으로 도서목록](#)

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

- DB 값이 제대로 출력 되었는지 HeidiSQL로 확인 했습니다.

Java1_bookstore



HeidiSQL

bookId	bookName	publisher	price
1	축구의 역사	굿스포츠	7,000
2	축구의 이해	대한미디어	22,000
3	골프바이블	대한미디어	35,000

3-1. 기능 구현

- Book의 목록, 등록, 수정, 삭제 기능 구현을 해보겠습니다.

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>book::register</title>
  </head>
  <body>
    <h3>도서등록</h3>

    <a href=" ../index.jsp">처음으로</a>
    <a href=" ../list.jsp">도서목록</a>

    <form action=" ../registerProc.jsp" method="post">
      <table border="1">
        <tr>
          <td>도서번호</td>
          <td><input type="number" name="bookId"></td>
        </tr>
        <tr>
          <td>도서명</td>
          <td><input type="text" name="bookName"></td>
        </tr>
        <tr>
          <td>출판사</td>
          <td><input type="text" name="publisher"></td>
        </tr>
        <tr>
          <td>가격</td>
          <td><input type="number" name="price"></td>
        </tr>
        <tr>
          <td colspan="2" align="right">
            <input type="submit" value="등록">
          </td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

```
<%@page import="java.sql.PreparedStatement"%>
<%@page import="config.JDBC"%>
<%@page import="java.sql.Connection"%>
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%
  request.setCharacterEncoding("UTF-8");

  int bookId = Integer.parseInt(request.getParameter("bookId"));
  String bookName = request.getParameter("bookName");
  String publisher = request.getParameter("publisher");
  int price = Integer.parseInt(request.getParameter("price"));

  try{
    Connection conn = JDBC.getInstance().getConnection();

    String sql = "INSERT INTO `book` VALUES (?, ?, ?, ?)";

    PreparedStatement psmt = conn.prepareStatement(sql);
    psmt.setInt(1, bookId);
    psmt.setString(2, bookName);
    psmt.setString(3, publisher);
    psmt.setInt(4, price);

    psmt.executeUpdate();

    psmt.close();
    conn.close();

  }catch(Exception e){
    e.printStackTrace();
  }

  response.sendRedirect("../list.jsp");
%>
```

처음으로 도서목록

도서번호	4
도서명	정보처리기사
출판사	대한미디어
가격	25000
<input type="button" value="등록"/>	

도서목록

처음으로 도서등록

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구의 이해	대한미디어	22000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	정보처리기사	대한미디어	25000	수정 삭제

3-2. 기능 구현

- Book의 목록, 등록, 수정, 삭제 기능 구현을 해보겠습니다.

```
<%@page import="java.sql.ResultSet"%>
<%@page import="java.sql.Statement"%>
<%@page import="bean.BookBean"%>
<%@page import="config.JDBC"%>
<%@page import="java.sql.Connection"%>
<%@page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%
    request.setCharacterEncoding("UTF-8");

    String bookId = request.getParameter("bookId");

    BookBean bk = null;

    try{
        Connection conn = JDBC.getInstance().getConnection();
        Statement stmt = conn.createStatement();

        String sql = "SELECT * FROM `book` WHERE `bookId`='"+bookId+"'";
        ResultSet rs = stmt.executeQuery(sql);

        bk = new BookBean();

        if(rs.next()){
            bk.setBookId(rs.getInt(1));
            bk.setBookName(rs.getString(2));
            bk.setPublisher(rs.getString(3));
            bk.setPrice(rs.getInt(4));
        }

        rs.close();
        stmt.close();
        conn.close();
    }catch(Exception e){
        e.printStackTrace();
    }
%>
```

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>book:modify</title>
</head>
<body>
    <h3>도서수정</h3>

    <a href=".."../index.jsp">처음으로</a>
    <a href=".."../list.jsp">도서목록</a>

    <form action=".."../modifyProc.jsp" method="post">
        <table border="1">
            <tr>
                <td>도서번호</td>
                <td><input type="number" name="bookId"></td>
            </tr>
            <tr>
                <td>도서명</td>
                <td><input type="text" name="bookName"></td>
            </tr>
            <tr>
                <td>출판사</td>
                <td><input type="text" name="publisher"></td>
            </tr>
            <tr>
                <td>가격</td>
                <td><input type="number" name="price"></td>
            </tr>
            <tr>
                <td colspan="2" align="right">
                    <input type="submit" value="수정">
                </td>
            </tr>
        </table>
    </form>
</body>
</html>
```

```
<%@page import="java.sql.PreparedStatement"%>
<%@page import="config.JDBC"%>
<%@page import="java.sql.Connection"%>
<%@page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%
    request.setCharacterEncoding("UTF-8");

    int bookId = Integer.parseInt(request.getParameter("bookId"));
    String bookName = request.getParameter("bookName");
    String publisher = request.getParameter("publisher");
    int price = Integer.parseInt(request.getParameter("price"));

    try{
        Connection conn = JDBC.getInstance().getConnection();
        String sql = "UPDATE `book` SET `bookName`=?, `publisher`=?, `price`=? ";
        sql += "WHERE `bookId`=?";

        PreparedStatement psmt = conn.prepareStatement(sql);

        psmt.setString(1, bookName);
        psmt.setString(2, publisher);
        psmt.setInt(3, price);
        psmt.setInt(4, bookId);

        psmt.executeUpdate();

        psmt.close();
        conn.close();
    }catch(Exception e){
        e.printStackTrace();
    }

    response.sendRedirect(".."../list.jsp");
%>
```

3-3. 기능 구현

- Book의 목록, 등록, 수정, 삭제 기능 구현을 해보겠습니다.

도서수정

[처음으로 도서목록](#)

도서번호	3
도서명	자바프로그래밍 입문
출판사	대한미디어
가격	55000
<input type="button" value="수정"/>	

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구의 이해	대한미디어	22000	수정 삭제
3	자바프로그래밍 입문	대한미디어	55000	수정 삭제

3-4. 기능 구현

- Book의 목록, 등록, 수정, 삭제 기능 구현을 해보겠습니다.

도서목록

[처음으로](#) [도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구의 이해	대한미디어	22000	수정 삭제

끝