

모던 웹 애플리케이션 개발2

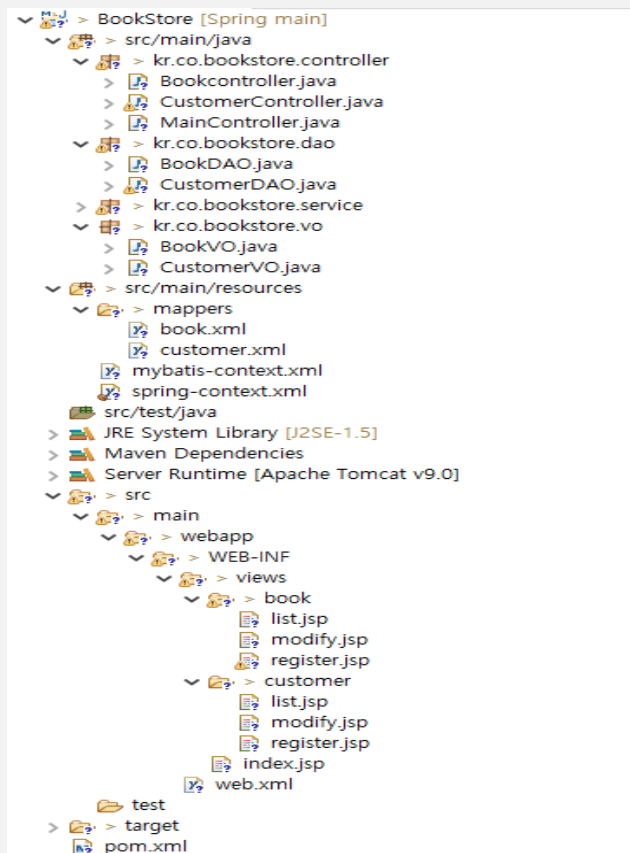
2023/01/04

임민지

목 차

- 1. 프로젝트 생성 및 구성
- 2. 화면 구현
- 3. 기능 구현
- 4. 실행

문제 1. 프로젝트 생성 및 구성



- ProjectName : BookStore
- WAS : Tomcat 9
- DB : javaI_bookstore
- Framework : Spring, Mybatis
- Tools : Eclipse, workbench

문제 2. 화면 구현

BOOK 목록

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨교본	굿스포츠	8000	수정 삭제
6	역도 단계 별기술	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	jjj	jjjjji	1200	수정 삭제
23	i□~ê³¼ë°□ë□□ë□□	i□m□´	8000	수정 삭제
34	sdfgdfsg	dsfgs	3333	수정 삭제
123	sdf	sgetryhty	777777	수정 삭제
555	i□~ê³¼	i□~ê³¼i\$□	5000	수정 삭제
777	sdfs	sdf	1000	수정 삭제
999	etrtwr	wertrt	888999	수정 삭제
5656	ë~,ë~,ë~,	ë\$□ë\$□ë\$□	5000	수정 삭제
546546	jjjjjk	wttwew	1111	수정 삭제

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Book::list</title>
  </head>
  <body>
    <h3>도서목록</h3>
    <a href="/BookStore/">처음으로</a>
    <a href="/BookStore/book/register">도서등록</a>

    <table border="1">
      <tr>
        <th>도서번호</th>
        <th>도서명</th>
        <th>출판사</th>
        <th>가격</th>
        <th>관리</th>
      </tr>
      <c:forEach var="book" items="${books}">
        <tr>
          <td>${book.bookId}</td>
          <td>${book.bookName}</td>
          <td>${book.publisher}</td>
          <td>${book.price}</td>
          <td>
            <a href="/BookStore/book/modify?uid=${book.bookId}">수정</a>
            <a href="/BookStore/book/delete?uid=${book.bookId}">삭제</a>
          </td>
        </tr>
      </c:forEach>
    </table>
  </body>
</html>
```

문제 2. 화면 구현

BOOK 등록

← → ↻ ⓘ localhost:8080/BookStore/book/register

[처음으로](#) [도서목록](#)

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

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>도서등록</title>
  </head>
  <body>
    <a href="/BookStore">처음으로</a>
    <a href="/BookStore/book/List">도서목록</a>

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

문제 2. 화면 구현

BOOK 수정

← → ↻ ⓘ localhost:8080/BookStore/book/modify?bookId=1

도서수정

[처음으로 도서목록](#)

도서번호	<input type="text" value="1"/>
도서명	<input type="text" value="축구의 역사"/>
출판사	<input type="text" value="굿스포츠"/>
가격	<input type="text" value="7000"/>
<input type="button" value="수정"/>	

```
<%@ page contentType="text/html;charset=UTF-8" pageEncoding="UTF-8"%%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Customer::list</title>
</head>
<body>
<h3>고객목록</h3>
<a href="/BookStore/">처음으로</a>
<a href="/BookStore/customer/register">고객등록</a>

<table border="1">
<tr>
<th>고객번호</th>
<th>고객명</th>
<th>주소</th>
<th>휴대폰</th>
<th>관리</th>
</tr>
<c:forEach var="customer" items="${customers}">
<tr>
<td>${customer.custId}</td>
<td>${customer.name}</td>
<td>${customer.address}</td>
<td>${customer.phone}</td>
<td>
<a href="/BookStore/customer/modify?custId=${customer.custId}">수정</a>
<a href="/BookStore/customer/delete?custId=${customer.custId}">삭제</a>
</td>
</tr>
</c:forEach>
</table>
</body>
</html>
```

문제 2. 화면 구현

CUSTOMER 목록

← → ↻ ⓘ localhost:8080/BookStore/customer/list

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

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Customer::modify</title>
  </head>
  <body>
    <h3>고객수정</h3>
    <a href="/BookStore/">처음으로</a>
    <a href="/BookStore/customer/List">고객목록</a>

    <form action="/BookStore/customer/modify" method="post">
      <table border="1">
        <tr>
          <th>고객번호</th>
          <td><input type="text" name="custId" value="${customer.custId}" readonly="readonly"></td>
        </tr>
        <tr>
          <th>고객명</th>
          <td><input type="text" name="name" value="${customer.name}"></td>
        </tr>
        <tr>
          <th>주소</th>
          <td><input type="text" name="address" value="${customer.address}"></td>
        </tr>
        <tr>
          <th>휴대폰</th>
          <td><input type="text" name="phone" value="${customer.phone}"></td>
        </tr>
        <tr>
          <td colspan="2" align="right"><input type="submit" value="수정"></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

문제 2. 화면 구현

CUSTOMER 등록

← → ↻ ⓘ localhost:8080/BookStore/customer/register

고객등록

[처음으로 고객목록](#)

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

```
<? page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="UTF-8">
    <title>Customer::modify</title>
  </head>
  <body>
    <h3>고객수정</h3>
    <a href="/BookStore/">처음으로</a>
    <a href="/BookStore/customer/register">고객목록</a>

    <form action="/BookStore/customer/modify" method="post">
      <table border="1">
        <tr>
          <th>고객번호</th>
          <td><input type="text" name="custId" value="${customer.custId}" readonly="readonly"></td>
        </tr>
        <tr>
          <th>고객명</th>
          <td><input type="text" name="name" value="${customer.name}"></td>
        </tr>
        <tr>
          <th>주소</th>
          <td><input type="text" name="address" value="${customer.address}"></td>
        </tr>
        <tr>
          <th>휴대폰</th>
          <td><input type="text" name="phone" value="${customer.phone}"></td>
        </tr>
        <tr>
          <td colspan="2" align="right"><input type="submit" value="수정"></td>
        </tr>
      </table>
    </form>
  </body>
</html>
```


문제 2. 화면 구현

CUSTOMER 수정

← → ↻ ⓘ localhost:8080/BookStore/customer/modify?custId=1

고객수정

[처음으로](#) [고객등록](#)

고객번호	1
고객명	박지성
주소	영국 맨체스터
휴대폰	000-5000-0001
<input type="button" value="수정"/>	

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Customer::list</title>
</head>
<body>
<h3>고객목록</h3>
<a href="/BookStore/">처음으로</a>
<a href="/BookStore/customer/register">고객등록</a>

<table border="1">
<tr>
<th>고객번호</th>
<th>고객명</th>
<th>주소</th>
<th>휴대폰</th>
<th>관리</th>
</tr>
<c:forEach var="customer" items="${customers}">
<tr>
<td>${customer.custId}</td>
<td>${customer.name}</td>
<td>${customer.address}</td>
<td>${customer.phone}</td>
<td>
<a href="/BookStore/customer/modify?custId=${customer.custId}">수정</a>
<a href="/BookStore/customer/delete?custId=${customer.custId}">삭제</a>
</td>
</tr>
</c:forEach>
</table>
</body>
</html>
```

문제 3. 기능 구현

BOOKDAO

```
package kr.co.bookstore.dao;

import java.util.List;

@Repository
public class BookDAO {

    @Autowired
    private SqlSessionTemplate mybatis;

    public void insertBook(BookVO vo) {
        mybatis.insert("book.insertBook", vo);
    }

    public BookVO selectBook(int bookId) {
        return mybatis.selectOne("book.selectBook", bookId);
    }

    public List<BookVO> selectBooks() {
        return mybatis.selectList("book.selectBooks");
    }

    public void updateBook(BookVO vo) {
        mybatis.update("book.updateBook", vo);
    }

    public void deleteBook(int bookId) {
        mybatis.delete("book.deleteBook", bookId);
    }
}
```

BOOKCONTROLLER

```
package kr.co.bookstore.controller;

import java.util.List;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.stereotype.Controller;
import org.springframework.ui.Model;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.PostMapping;

import kr.co.bookstore.service.BookService;
import kr.co.bookstore.vo.BookVO;

@Controller
public class Bookcontroller {

    @Autowired
    private BookService service;

    @GetMapping("/book/list")
    public String list(Model model) {
        List<BookVO> books = service.selectBooks();
        model.addAttribute("books", books);
        return "/book/list";
    }

    @GetMapping("/book/register")
    public String register() {
        return "/book/register";
    }

    @PostMapping("/book/register")
    public String register(BookVO vo) {
        service.insertBook(vo);
        return "redirect:/book/list";
    }

    @PostMapping("/book/modify")
    public String modify(BookVO vo) {
        service.updateBook(vo);
        return "redirect:/book/list";
    }

    @GetMapping("/book/modify")
    public String modify(Model model, int bookId) {
        BookVO book = service.selectBook(bookId);
        model.addAttribute("book", book);
        return "/book/modify";
    }

    @GetMapping("/book/delete")
    public String delete(int bookId) {
        service.deleteBook(bookId);
        return "redirect:/book/list";
    }
}
```

BOOKVO

```
public class BookVO {

    private int bookId;
    private String bookName;
    private String publisher;
    private int price;

    public int getBookId() {
        return bookId;
    }

    public void setBookId(int bookId) {
        this.bookId = bookId;
    }

    public void setBookId(String bookId) {
        this.bookId = Integer.parseInt(bookId);
    }

    public String getBookName() {
        return bookName;
    }

    public void setBookName(String bookName) {
        this.bookName = bookName;
    }

    public String getPublisher() {
        return publisher;
    }

    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }

    public int getPrice() {
        return price;
    }

    public void setPrice(int price) {
        this.price = price;
    }

    public void setprice(String price) {
        this.price = Integer.parseInt(price);
    }
}
```

문제 3. 기능 구현

CUSTOMERDAO

```
package kr.co.bookstore.dao;

import java.util.List;

@Repository
public class CustomerDAO {

    @Autowired
    private SqlSessionTemplate mybatis;

    public void insertCustomer(CustomerVO vo) {
        mybatis.insert("customer.insertCustomer", vo);
    }

    public CustomerVO selectCustomer(int custId) {
        return mybatis.selectOne("customer.selectCustomer", custId);
    }

    public List<CustomerVO> selectCustomers() {
        return mybatis.selectList("customer.selectCustomers");
    }

    public void updateCustomer(CustomerVO vo) {
        mybatis.update("customer.updateCustomer", vo);
    }

    public void deleteCustomer(int custId) {
        mybatis.delete("customer.deleteCustomer", custId);
    }
}
```

CUSTOMERCONTROLLER

```
import java.util.List;

@Controller
public class CustomerController {

    @Autowired
    private CustomerService service;

    @GetMapping("/customer/list")
    public String list(Model model) {
        List<CustomerVO> customers = service.selectCustomers();
        model.addAttribute("customers", customers);
        return "/customer/list";
    }

    @GetMapping("/customer/register")
    public String register() {
        return "/customer/register";
    }

    @PostMapping("/customer/register")
    public String register(CustomerVO vo) {
        service.insertCustomer(vo);
        return "redirect:/customer/list";
    }

    @GetMapping("/customer/modify")
    public String modify(Model model, int custId) {
        CustomerVO customer = service.selectCustomer(custId);
        model.addAttribute("customer", customer);
        return "/customer/modify";
    }

    @PostMapping("/customer/modify")
    public String modify(CustomerVO vo) {
        service.updateCustomer(vo);
        return "redirect:/customer/list";
    }

    @GetMapping("/customer/delete")
    public String delete(int custId) {
        service.deleteCustomer(custId);
        return "redirect:/customer/list";
    }
}
```

CUSTOMERVO

```
package kr.co.bookstore.vo;

public class CustomerVO {

    public int custId;
    public String name;
    public String address;
    public String phone;

    public int getCustId() {
        return custId;
    }
    public void setCustId(int custId) {
        this.custId = custId;
    }
    public void setCustId(String custId) {
        this.custId = Integer.parseInt(custId);
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getAddress() {
        return address;
    }
    public void setAddress(String address) {
        this.address = address;
    }
    public String getPhone() {
        return phone;
    }
    public void setPhone(String phone) {
        this.phone = phone;
    }
}
```

문제 4. 실행

고객 목록 수정

고객목록

[처음으로](#) [고객등록](#)

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 강원도	000-6000-0001	수정 삭제
3	장미란	대한민국 서울	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	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 서울	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제

김연아의 주소를 대한민국 강원도 에서
대한민국 서울로 수정

문제 4. 실행

고객 목록 삭제

고객목록

[처음으로](#) [고객등록](#)

고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	수정 삭제
2	김연아	대한민국 강원도	000-6000-0001	수정 삭제
3	장미란	대한민국 서울	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제

← → ↻ ⓘ localhost:8080/BookStore/customer/list

고객목록

[처음으로](#) [고객등록](#)

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 강원도	000-6000-0001	수정 삭제
3	장미란	대한민국 서울	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제

고객목록에서 박지성 삭제

문제 4. 실행

책 목록 수정

도서목록				
처음으로 도서등록				
도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨교본	굿스포츠	8000	수정 삭제
6	역도 단계별기술	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	jjj	jjjjj	1200	수정 삭제
23	iQ~ê³¼e*QeQQeQQ	iQ=iQ´	8000	수정 삭제
34	sdfgdfsg	dsfgs	3333	수정 삭제
123	sdf	sgetryhty	777777	수정 삭제
555	iQ~ê³¼	iQ~ê³¼i\$Q	5000	수정 삭제
777	sdfs	sdf	1000	수정 삭제
999	etrtwr	wertrt	888999	수정 삭제
5656	ë",ë",ë",	ë\$Që\$Që\$Q	5000	수정 삭제
546546	jjjjjk	wttwew	1111	수정 삭제

← → ↺ localhost:8080/BookStore/book/list				
도서목록				
처음으로 도서등록				
도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	10000	수정 삭제
2	축구아는 여자	나무수	13000	수정 삭제
3	축구의 이해	대한미디어	22000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨교본	굿스포츠	8000	수정 삭제
6	역도 단계별기술	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	jjj	jjjjj	1200	수정 삭제
23	iQ~ê³¼e*QeQQeQQ	iQ=iQ´	8000	수정 삭제
34	sdfgdfsg	dsfgs	3333	수정 삭제
123	sdf	sgetryhty	777777	수정 삭제
555	iQ~ê³¼	iQ~ê³¼i\$Q	5000	수정 삭제
777	sdfs	sdf	1000	수정 삭제
999	etrtwr	wertrt	888999	수정 삭제
5656	ë",ë",ë",	ë\$Që\$Që\$Q	5000	수정 삭제
546546	jjjjjk	wttwew	1111	수정 삭제

축구의 역사 7000 > 10000 수정

