

정기수행평가

# 모던 웹 애플리케이션 개발2

2023/01/04

길은섭

# 목차

## 1. 프로젝트 생성 및 구성

## 2. 화면 구현

- book(도서) 목록, 등록, 수정 화면 구현
- customer(고객) 목록, 등록, 수정 화면 구현

## 3.기능 구현

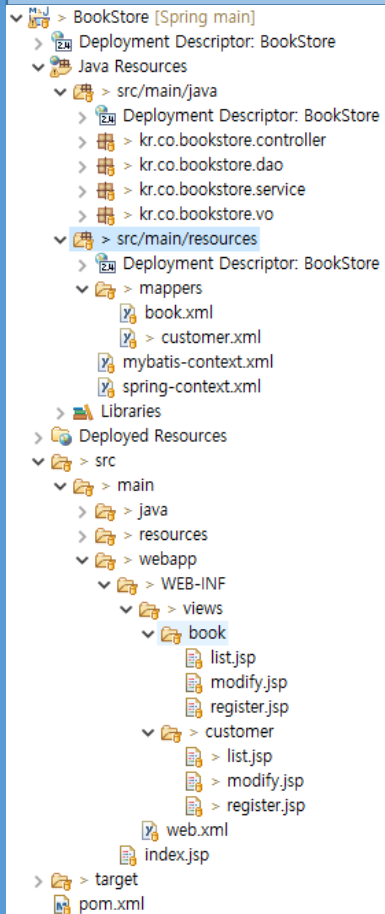
- book(도서) 기본 CRUD
- customer (고객) 기본 CRUD

## 3.기능 구현

- book 목록, 등록, 수정, 삭제 기능 동작 화면 첨부
- customer 목록, 등록, 수정, 삭제 기능 동작 화면 첨부

# 1. 프로젝트 생성 및 구성

## 프로젝트 구성



## 프로젝트에 사용된 프로그램

- **ProjectName** : BookStore
- **WAS** : Tomcat 9
- **DB** : java1\_bookstore
- **Framework** : Spring, Mybatis
- **Tools** : Eclipse, Workbench

## 2-1. 화면 구현 - Book(도서)

Index

index x +

localhost:8080/BookStore/

BookStore

[도서목록](#) [고객목록](#)

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	<a href="#">수정</a> <a href="#">삭제</a>
2	축구의 이해	대한미디어	22000	<a href="#">수정</a> <a href="#">삭제</a>
3	골프바이블	대한미디어	35000	<a href="#">수정</a> <a href="#">삭제</a>

도서등록

도서등록

[처음으로 도서목록](#)

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

도서수정

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"/>	

## 2-2. 화면 구현 - Customer(고객)

고객목록				
고객목록				
<a href="#">처음으로 고객등록</a>				
고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스타	000-5000-0001	<a href="#">수정</a> <a href="#">삭제</a>
2	김연아	대한민국 서울	000-6000-0001	<a href="#">수정</a> <a href="#">삭제</a>
3	장미란	대한민국 강원도	000-7000-0001	<a href="#">수정</a> <a href="#">삭제</a>
4	추신수	미국 클리블랜드	000-8000-0001	<a href="#">수정</a> <a href="#">삭제</a>
5	박세리	대한민국 대전	000-9000-0001	<a href="#">수정</a> <a href="#">삭제</a>

고객등록	
고객등록	
<a href="#">처음으로 고객목록</a>	
고객번호	<input type="text"/>
고객명	<input type="text"/>
주소	<input type="text"/>
휴대폰	<input type="text"/>
<input type="button" value="등록"/>	

고객수정	
고객수정	
<a href="#">처음으로 고객목록</a>	
고객번호	<input type="text" value="1"/>
고객명	<input type="text" value="박지성"/>
주소	<input type="text" value="영국 맨체스타"/>
휴대폰	<input type="text"/>
<input type="button" value="수정"/>	

# 3-1. 기능 구현 – Book(도서)

## Controller

```
@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";
    }

    @GetMapping("/book/modify")
    public String modify(Model model, String bookId) {

        BookVO books = service.selectBook(bookId);
        model.addAttribute("books", books);

        return "/book/modify";
    }

    @PostMapping("/book/modify")
    public String modify(BookVO vo) {
        service.updateBook(vo);

        return "redirect:/book/list";
    }

    @GetMapping("/book/delete")
    public String delete(String bookId) {

        service.deleteBook(bookId);

        return "redirect:/book/list";
    }
}
```

## DAO

```
@Repository
public class BookDAO {

    @Autowired
    private SqlSessionTemplate mybatis;

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

    public BookVO selectBook(String uid) {
        return mybatis.selectOne("book.selectBook", uid);
    }

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

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

    public void deleteBook(String uid) {
        mybatis.delete("book.deleteBook", uid);
    }
}
```

## Query

```
<mapper namespace="book">
    <insert id="insertBook">
        INSERT INTO `book` VALUES ({bookId}, #{bookName}, #{publisher}, #{price});
    </insert>
    <select id="selectBook" resultType="kr.co.bookstore.vo.BookVO">
        SELECT * FROM `book` WHERE `bookId`=#{bookId};
    </select>
    <select id="selectBooks" resultType="kr.co.bookstore.vo.BookVO">
        SELECT * FROM `book`;
    </select>
    <update id="updateBook">
        UPDATE `book` SET
            `bookName`=#{bookName},
            `publisher`=#{publisher},
            `price`=#{price}
        WHERE
            `bookId`=#{bookId};
    </update>
    <delete id="deleteBook">
        DELETE FROM `book` WHERE `bookId`=#{bookId};
    </delete>
</mapper>
```

## 3-2. 기능 구현 – Customer(고객)

### Controller

```
@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, String custId) {
        CustomerVO customers = service.selectCustomer(custId);
        model.addAttribute("customers", customers);

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

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

        return "redirect:/customer/list";
    }
}
```

### DAO

```
@Repository
public class CustomerDAO {

    @Autowired
    private SqlSessionTemplate mybatis;

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

    public CustomerVO selectCustomer(String 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(String custId) {
        mybatis.delete("customer.deleteCustomer", custId);
    }
}
```

### Query

```
<mapper namespace="customer">
    <insert id="insertCustomer">
        INSERT INTO `customer` VALUES (#{custId}, #{name}, #{address}, #{phone});
    </insert>
    <select id="selectCustomer" resultType="kr.co.bookstore.vo.CustomerVO">
        SELECT * FROM `customer` WHERE `custId`=#{custId};
    </select>
    <select id="selectCustomers" resultType="kr.co.bookstore.vo.CustomerVO">
        SELECT * FROM `customer`;
    </select>
    <update id="updateCustomer">
        UPDATE `customer` SET
            `name`=#{name},
            `address`=#{address},
            `phone`=#{phone}
        WHERE
            `custId`=#{custId};
    </update>
    <delete id="deleteCustomer">
        DELETE FROM `customer` WHERE `custId`=#{custId};
    </delete>
</mapper>
```

## 4-1. 실행 - Book(도서)

### 도서목록

도서목록

[처음으로 도서등록](#)

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	<a href="#">수정</a> <a href="#">삭제</a>
2	축구의이해	대한미디어	22000	<a href="#">수정</a> <a href="#">삭제</a>
3	골프바이블	대한미디어	35000	<a href="#">수정</a> <a href="#">삭제</a>

### 도서등록

도서등록

[처음으로 도서목록](#)

도서번호	<input type="text" value="4"/>
도서명	<input type="text" value="정보처리기사 필기"/>
출판사	<input type="text" value="시나공"/>
가격	<input type="text" value="35000"/>
<input type="button" value="등록"/>	

### 도서등록 확인

도서번호	도서명	출판사	가격	관리
1	축구의 역사	굿스포츠	7000	<a href="#">수정</a> <a href="#">삭제</a>
2	축구의이해	대한미디어	22000	<a href="#">수정</a> <a href="#">삭제</a>
3	골프바이블	대한미디어	35000	<a href="#">수정</a> <a href="#">삭제</a>
4	정보처리기사 필기	시나공	35000	<a href="#">수정</a> <a href="#">삭제</a>

### 도서수정

도서수정

도서번호	<input type="text" value="1"/>
도서명	<input type="text" value="시물라시옹 시물라크르"/>
출판사	<input type="text" value="장 보드리야르"/>
가격	<input type="text" value="15000"/>
<input type="button" value="수정"/>	

### 도서수정 확인

도서번호	도서명	출판사	가격	관리
1	시물라시옹 시물라크르	장 보드리야르	15000	<a href="#">수정</a> <a href="#">삭제</a>
2	축구의이해	대한미디어	22000	<a href="#">수정</a> <a href="#">삭제</a>
3	골프바이블	대한미디어	35000	<a href="#">수정</a> <a href="#">삭제</a>
4	정보처리기사 필기	시나공	35000	<a href="#">수정</a> <a href="#">삭제</a>



## 4-2. 실행 - Customer(고객)

고객목록				
고객목록				
<a href="#">처음으로 고객등록</a>				
고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스터	000-5000-0001	<a href="#">수정</a> <a href="#">삭제</a>
2	김연아	대한민국 서울	000-6000-0001	<a href="#">수정</a> <a href="#">삭제</a>
3	장미란	대한민국 강원도	000-7000-0001	<a href="#">수정</a> <a href="#">삭제</a>
4	추신수	미국 클리블랜드	000-8000-0001	<a href="#">수정</a> <a href="#">삭제</a>
5	박세리	대한민국 대전	000-9000-0001	<a href="#">수정</a> <a href="#">삭제</a>

고객등록	
고객등록	
<a href="#">처음으로 고객목록</a>	
고객번호	6
고객명	길은섭
주소	대한민국 부산
휴대폰	010-1234-1001
<input type="button" value="등록"/>	

고객등록 확인				
고객등록				
<a href="#">처음으로 고객등록</a>				
고객번호	고객명	주소	휴대폰	관리
1	박지성	영국 맨체스터	000-5000-0001	<a href="#">수정</a> <a href="#">삭제</a>
2	김연아	대한민국 서울	000-6000-0001	<a href="#">수정</a> <a href="#">삭제</a>
3	장미란	대한민국 강원도	000-7000-0001	<a href="#">수정</a> <a href="#">삭제</a>
4	추신수	미국 클리블랜드	000-8000-0001	<a href="#">수정</a> <a href="#">삭제</a>
5	박세리	대한민국 대전	000-9000-0001	<a href="#">수정</a> <a href="#">삭제</a>
6	길은섭	대한민국 부산	010-1234-1001	<a href="#">수정</a> <a href="#">삭제</a>

고객수정	
고객수정	
<a href="#">처음으로 고객목록</a>	
고객번호	1
고객명	이순신
주소	대한민국 부산
휴대폰	010-1234-1002
<input type="button" value="수정"/>	

고객수정 확인				
고객수정				
<a href="#">처음으로 고객등록</a>				
고객번호	고객명	주소	휴대폰	관리
1	이순신	대한민국 부산	010-1234-1002	<a href="#">수정</a> <a href="#">삭제</a>
2	김연아	대한민국 서울	000-6000-0001	<a href="#">수정</a> <a href="#">삭제</a>
3	장미란	대한민국 강원도	000-7000-0001	<a href="#">수정</a> <a href="#">삭제</a>
4	추신수	미국 클리블랜드	000-8000-0001	<a href="#">수정</a> <a href="#">삭제</a>
5	박세리	대한민국 대전	000-9000-0001	<a href="#">수정</a> <a href="#">삭제</a>
6	길은섭	대한민국 부산	010-1234-1001	<a href="#">수정</a> <a href="#">삭제</a>

**감사합니다**