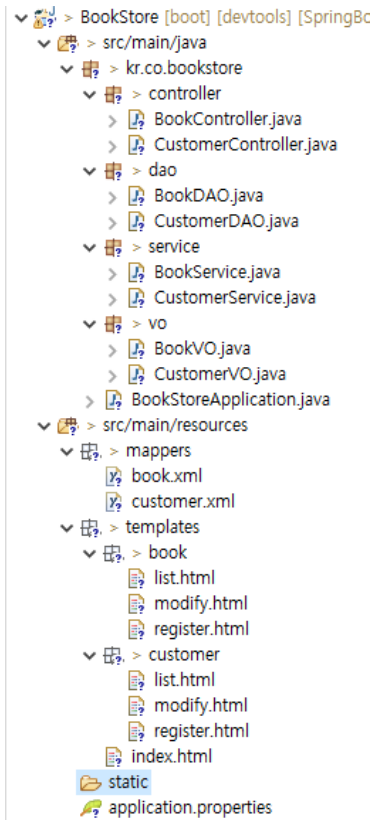


서버 프로그램 구현

2023/01/11 라성준

문제1. 프로젝트 생성 및 구성



- ▶ ProjectName : BookStore
- ▶ WAS : Tomcat 9
- ▶ DB : java2_bookstore
- ▶ Framework : Spring(SpringBoot), Mybatis
- ▶ Tools : Eclipse, Workbench

문제2. 화면 구현

book 목록

메인 도서 등록

도서번호	도서명	출판사	가격	관리
1	자바의 정석	굿스포츠	7000	수정 삭제
2	데이터베이스론	영진출판사	30000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	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
222	테스트도서	테스트	333	수정 삭제

도서 수정

메인 도서 목록

도서번호	222
도서명	테스트도서
출판사	테스트
가격	333
<input type="button" value="수정"/>	

도서 등록

메인 도서 목록

도서번호	0
도서명	
출판사	
가격	0
<input type="button" value="등록"/>	

customer 목록

메인 고객 등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	010-5555-6001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	010-1234-1111	수정 삭제
7	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
8	장미란	대한민국 강원도	000-7000-0001	수정 삭제
9	김연아	대한민국 서울	-6001	수정 삭제
10	박지성	영국 맨체스타	000-5000-0001	수정 삭제
11	박세리	대한민국 대전		수정 삭제

고객 등록

메인 고객 목록

고객번호	0
고객명	
주소	
휴대폰	
<input type="button" value="등록"/>	

문제3 기능구현

```
1 server.servlet.context-path=/BookStore
2 server.port=8080
3 spring.thymeleaf.cache=false
4
5 # Mybatis 설정
6 spring.datasource.url=jdbc:mysql://127.0.0.1:3306/java2_bookstore
7 spring.datasource.username=root
8 spring.datasource.password=1234
9 spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
10
11 #MyBatis Mapper 경로설정
12 mybatis.mapper-locations=classpath:mappers/**/*.xml
```

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

```
import lombok.AllArgsConstructor;...
```

```
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
@ToString
```

```
public class BookVO {
    private int bookId;
    private String bookname;
    private String publisher;
    private int price;
}
```

```
}
```

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

```
import lombok.AllArgsConstructor;...
```

```
@Getter
@Setter
@AllArgsConstructor
@NoArgsConstructor
@ToString
```

```
public class CustomerVO {
    private int custId;
    private String name;
    private String address;
    private String phone;
}
```

```
}
```

► Book ,customer vo

문제3 기능구현

```
-//mybatis.org//DTD Mapper 3.0//EN (doctype)
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper
  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
  "https://mybatis.org/dtd/mybatis-3-mapper.dtd">
<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>
```

▶ Book ,customer xml 설정

```
-//mybatis.org//DTD Mapper 3.0//EN (doctype)
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE mapper
  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
  "https://mybatis.org/dtd/mybatis-3-mapper.dtd">
<mapper namespace="customer">
  <insert id="insertCustomer">
    INSERT INTO `customer` (`name`, `address`, `phone`)VALUES (#{name}, #{address}, #{phone});
  </insert>
  <select id="selectCustomer" resultType="kr.co.bookstore.vo.CustomerVO">
    SELECT * FROM `customer` WHERE `customerId`=#{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 `bookId`=#{custId};
  </delete>
</mapper>
```

문제3 기능구현

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

import java.util.List;

@Controller
@RequestMapping
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(Model model, BookVO book) {
        model.addAttribute("book", book);
        return "/book/register";
    }

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

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

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

import java.util.List;

@Controller
@RequestMapping
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(Model model, CustomerVO customer) {
        model.addAttribute("customer", customer);
        return "/customer/register";
    }

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

    @GetMapping("/customer/modify")
    public String modify(String custId, Model model) {
        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(String custId) {
        service.deleteCustomer(custId);
        return "redirect:/customer/list";
    }
}
```

► Book ,customer controller

문제3 기능구현

```
package kr.co.bookstore.service;

import java.util.List;

@Service
public class BookService {

    @Autowired
    private BookDAO dao;

    public void insertBook(BookVO vo) {
        dao.insertBook(vo);
    }

    public BookVO selectBook(String bookId) {
        return dao.selectBook(bookId);
    }

    public List<BookVO> selectBooks() {
        return dao.selectBooks();
    }

    public void updateBook(BookVO vo) {
        dao.updateBook(vo);
    }

    public void deleteBook(String bookId) {
        dao.deleteBook(bookId);
    }
}
```

```
package kr.co.bookstore.service;

import java.util.List;

@Service
public class CustomerService {

    @Autowired
    private CustomerDAO dao;

    public void insertCustomer(CustomerVO vo) {
        dao.insertCustomer(vo);
    }

    public CustomerVO selectCustomer(String custId) {
        return dao.selectCustomer(custId);
    }

    public List<CustomerVO> selectCustomers() {
        return dao.selectCustomers();
    }

    public void updateCustomer(CustomerVO vo) {
        dao.updateCustomer(vo);
    }

    public void deleteCustomer(String custId) {
        dao.deleteCustomer(custId);
    }
}
```

▶ Book ,customer service

```

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(String 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(String bookId) {
        mybatis.delete("book.deleteBook", bookId);
    }
}

```

```

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(String custId) {
        return mybatis.selectOne("customer.selectBook", 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);
    }
}

```

► Book ,customer DAO

문제4 실행

book 목록

메인 도서 등록

도서번호	도서명	출판사	가격	관리
1	자바의 정석	굿스포츠	7000	수정 삭제
2	데이터베이스론	영진출판사	30000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	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
222	테스트도서	테스트	333	수정 삭제

▶ 도서 목록

도서 등록

메인 도서 목록

도서번호	9
도서명	Spring
출판사	스프링
가격	3000
<input type="button" value="등록"/>	

▶ 도서 등록

book 목록

메인 도서 등록

도서번호	도서명	출판사	가격	관리
1	자바의 정석	굿스포츠	7000	수정 삭제
2	데이터베이스론	영진출판사	30000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	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	Spring	스프링	3000	수정 삭제
222	테스트도서	테스트	333	수정 삭제

▶ 도서 등록 후 목록

도서 수정

메인 도서 목록

도서번호	9
도서명	Spring수정
출판사	스프링수정
가격	5000
<input type="button" value="수정"/>	

▶ 도서 수정

book 목록

메인 도서 등록

도서번호	도서명	출판사	가격	관리
1	자바의 정석	굿스포츠	7000	수정 삭제
2	데이터베이스론	영진출판사	30000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	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	Spring수정	스프링수정	5000	수정 삭제
222	테스트도서	테스트	333	수정 삭제

▶ 수정 후 목록

book 목록

메인 도서 등록

도서번호	도서명	출판사	가격	관리
1	자바의 정석	굿스포츠	7000	수정 삭제
2	데이터베이스론	영진출판사	30000	수정 삭제
3	골프바이블	대한미디어	35000	수정 삭제
4	골프 바이블	대한미디어	35000	수정 삭제
5	피겨 교본	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
222	테스트도서	테스트	333	수정 삭제

▶ 도서 삭제

문제4 실행

customer 목록

메인 고객 등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	010-5555-6001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	010-1234-1111	수정 삭제
7	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
8	장미란	대한민국 강원도	000-7000-0001	수정 삭제
9	김연아	대한민국 서울	-6001	수정 삭제
10	박지성	영국 맨체스터	000-5000-0001	수정 삭제

▶ 고객 목록

고객 등록

메인 고객 목록

고객명	<input type="text" value="라성준"/>
주소	<input type="text" value="창원시 진해구"/>
휴대폰	<input type="text" value="010-0111-0111"/>
<input type="button" value="등록"/>	

▶ 고객 등록

customer 목록

메인 고객 등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	010-5555-6001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	010-1234-1111	수정 삭제
7	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
8	장미란	대한민국 강원도	000-7000-0001	수정 삭제
9	김연아	대한민국 서울	-6001	수정 삭제
10	박지성	영국 맨체스터	000-5000-0001	수정 삭제
23	라성준	창원시 진해구	010-0111-0111	수정 삭제

▶ 고객 등록 후 목록

member 수정

메인 고객 목록

아이디	<input type="text" value="23"/>
이름	<input type="text" value="라성준"/>
주소	<input type="text" value="수정"/>
주소	<input type="text" value="010-0001-0111"/>
<input type="button" value="수정"/>	

▶ 고객 수정

customer 목록

메인 고객 등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	010-5555-6001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	010-1234-1111	수정 삭제
7	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
8	장미란	대한민국 강원도	000-7000-0001	수정 삭제
9	김연아	대한민국 서울	-6001	수정 삭제
10	박지성	영국 맨체스터	000-5000-0001	수정 삭제
23	라성준	수정	010-0001-0111	수정 삭제

▶ 고객 수정 후 목록

customer 목록

메인 고객 등록

고객번호	고객명	주소	휴대폰	관리
2	김연아	대한민국 서울	010-5555-6001	수정 삭제
3	장미란	대한민국 강원도	000-7000-0001	수정 삭제
4	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
5	박세리	대한민국 대전	010-1234-1111	수정 삭제
7	추신수	미국 클리블랜드	000-8000-0001	수정 삭제
8	장미란	대한민국 강원도	000-7000-0001	수정 삭제
9	김연아	대한민국 서울	-6001	수정 삭제
10	박지성	영국 맨체스터	000-5000-0001	수정 삭제

▶ 고객 삭제

감사합니다!