

- ❖ 학습해야 할 내용
- ✓ Spring@MVC
- ✓ Interceptor

S 전자 회사는 사회 환원을 목적으로 누구나 활용할 수 있는 오픈 도서관을 운영하고자 한다. 도서관리 프로젝트를 Spring MVC를 이용하여 구현 하여 보자.

데이터베이스 테이블은 다음과 같다. (DB 연동은 알맞게 구현한다)

```
create table book(
isbn char(12) primay key,
title varchar(100) not null,
catalogue varchar(50) not null,
nation varchar(50),
publish_date date,
publisher varchar(100),
author varchar(100) not null,
price int(6),
currency char(6),
description varchar(4000)
);
INSERT INTO book VALUES('1233-111-111','Java 완성','프로그래밍','국내도서',
         '2019.1.1','사무국','사무국',3000,'원','java를 마시자');
INSERT INTO book VALUES('1233-111-123', 'Java와 함께', '프로그래밍', '국내도서',
         '2019.1.2', '사무국', '사무국',4000, '원', 'java와 동행');
```



다만, 다음페이지에서 요구되는 내용을 Spring MVC에 맞게 구현해 올바른 결과를 제공해야 한다.

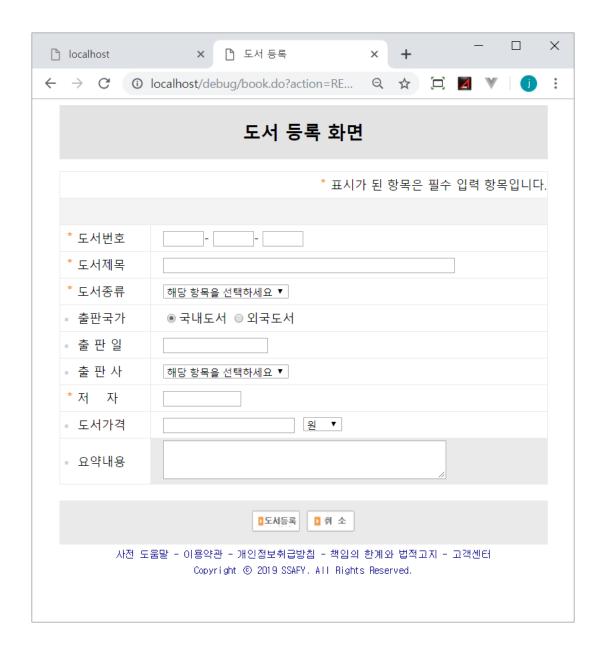
참고용으로 제공되는 작업 순서이다.

- 1. MySQL 테이블을 생성한다.
- 2. DB 연결을 확인한다.
  src/main/resources/config.properties 의 DB명과 id,pw 를 확인한다.
- 3. 구현해야 할 기능을 확인 한다.
- 4. DTO 클래스를 생성한다.
- 5. 필요한 SQL 쿼리를 작성한다.
- 6. Repository, Service, Controller클래스를 작성한다. src/main/java/ 에 작성한다.
- 7. 위 6번의 클래스들을 Spring Framework 에 알맞게 등록한다.
- 8. JSP나 html 파일을 작성하여 요구사항을 반영한다. src/main/webapp/내에 작성하고, Controller 클래스를 호출하여 실행한다.



#### | 구현해야 할 내용

도서 관리를 위한 도서 등록, 검색, 상세조회, 삭제 기능을 처리 하려고 한다. 아래의 실행결과와 같은 구조가 되도록 프로젝트를 작성하세요.



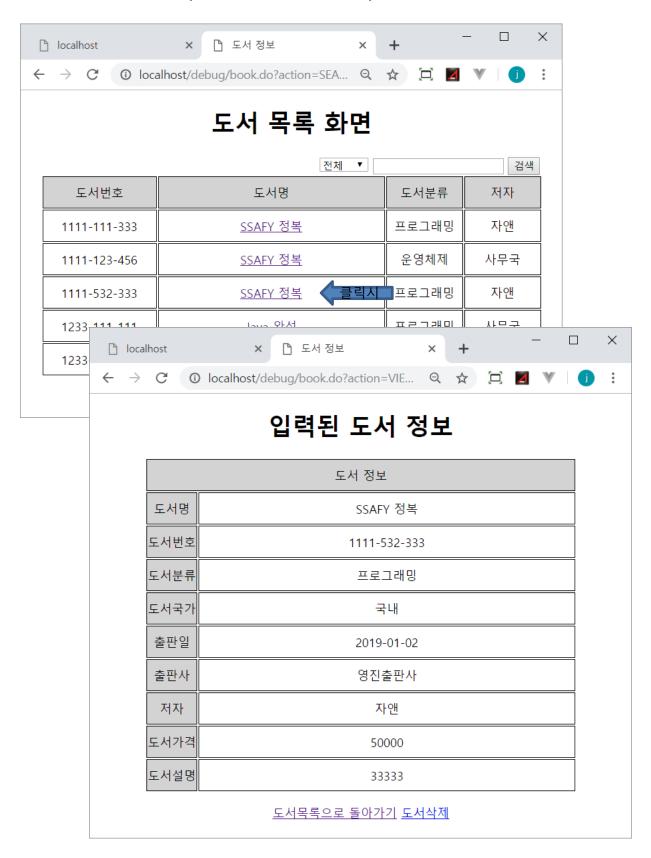


#### ▶ 도서 등록





▶ 도서 목록과 상세 조회 (추가기능 : 도서명 검색)





#### ▶ 도서 삭제 : 삭제 버튼 클릭시

