

Spring MyBatis

BookStore SELECT One by MyBatis

| MyBatis 이용하여 C R U D 처리

book_SQL.xml

parameterType	hashMap
resultType	hashMap

```
<select id="select_detail" parameterType="hashMap" resultType="hashMap">
  <![CDATA[
    SELECT
      title, category, price, insert_date
    FROM book
    WHERE
      book_id = #{bookId}
  ]]>
</select>
```

Class

BookRepository

```
Map<String, Object> selectDetail(Map<String, Object> map);
```

BookRepositoryImpl

selectOne

쿼리 행 수 0 이면 null 반환

결과 여러개면 예외 발생

한개만 !

```
@Override
public Map<String, Object> selectDetail(Map<String, Object> map) {
    return this.sqlSessionTemplate.selectOne("book.select_detail", map);
}
```

BookService

```
Map<String, Object> detail(Map<String, Object> map);
```

BookServiceImpl

```
@Override
public Map<String, Object> detail(Map<String, Object> map) {

    return this.bookRepository.selectDetail(map);
}
```

BookController

```
@GetMapping("/detail")
public ModelAndView detail(@RequestParam Map<String, Object> map) {
    Map<String, Object> detailMap = this.bookService.detail(map);

    ModelAndView mav = new ModelAndView();
    mav.addObject("data", detailMap);
    String bookId = map.get("bookId").toString();
    mav.addObject("bookId", bookId);
    mav.setViewName("book/detail");

    return mav;
}
```

View

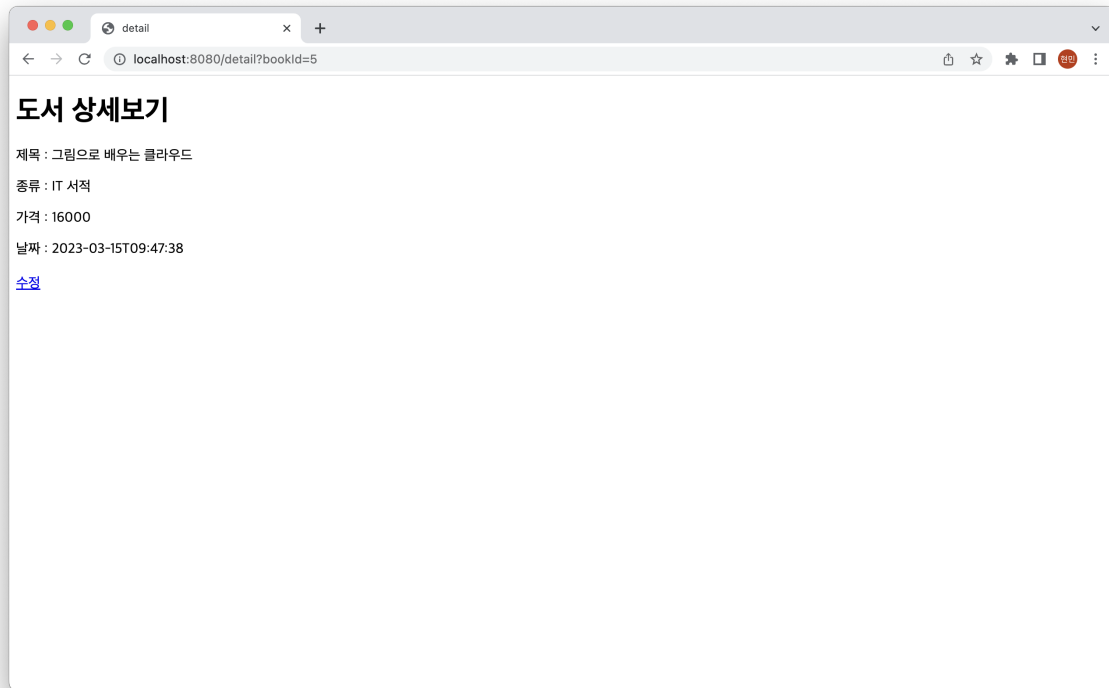
detail.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>detail</title>
</head>
<body>
    <h1>도서 상세보기</h1>
    <p>제목 : ${data.title }
    <p>종류 : ${data.category }
    <p>가격 : ${data.price }
    <p>날짜 : ${data.insert_date }
```

```

<br>
<br>
<a href="/update?bookId=${bookId}">수정</a>
</body>
</html>

```



BookStore UPDATE by MyBatis

| MyBatis 이용하여 C R **U** D 처리

book_SQL.xml

```

<update id="update" parameterType="hashMap">
  <![CDATA[
    UPDATE book
    SET
      title = #{title}, category = #{category}, price = #{price}
    WHERE
      book_id = #{bookId}
  ]>
</update>

```

Class

BookRepository

```
int update(Map<String, Object> map);
```

BookRepositoryImpl

```
@Override
public int update(Map<String, Object> map) {

    return this.sqlSessionTemplate.update("book.update", map);
}
```

BookService

```
boolean update(Map<String, Object> map);
```

BookServiceImpl

```
@Override
public boolean update(Map<String, Object> map) {

    int affectRowCount = this.bookRepository.update(map);

    return affectRowCount == 1;
}
```

BookController

```
@PostMapping("/update")
public ModelAndView updatePost(@RequestParam Map<String, Object> map) {

    ModelAndView mav = new ModelAndView();

    boolean isUpdateSuccess = this.bookService.update(map);
    if(isUpdateSuccess) {
        String bookId = map.get("bookId").toString();
        mav.setViewName("redirect:/detail?bookId=" + bookId);
    } else {
        mav = this.update(map);
    }
    return mav;
}
```

View

update.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>create</title>
</head>
<body>
    <h1>도서 수정</h1>
    <form method="post">
        <p>
            제목 : <input type="text" name="title" value="${data.title }"/>
        <p>
            종류 : <input type="text" name="category" value="${data.category }"/>
        <p>
            가격 : <input type="text" name="price" value="${data.price }"/>
        <p>
            <input type="submit" value="저장" />
    </form>
</body>
</html>
```

도서 수정

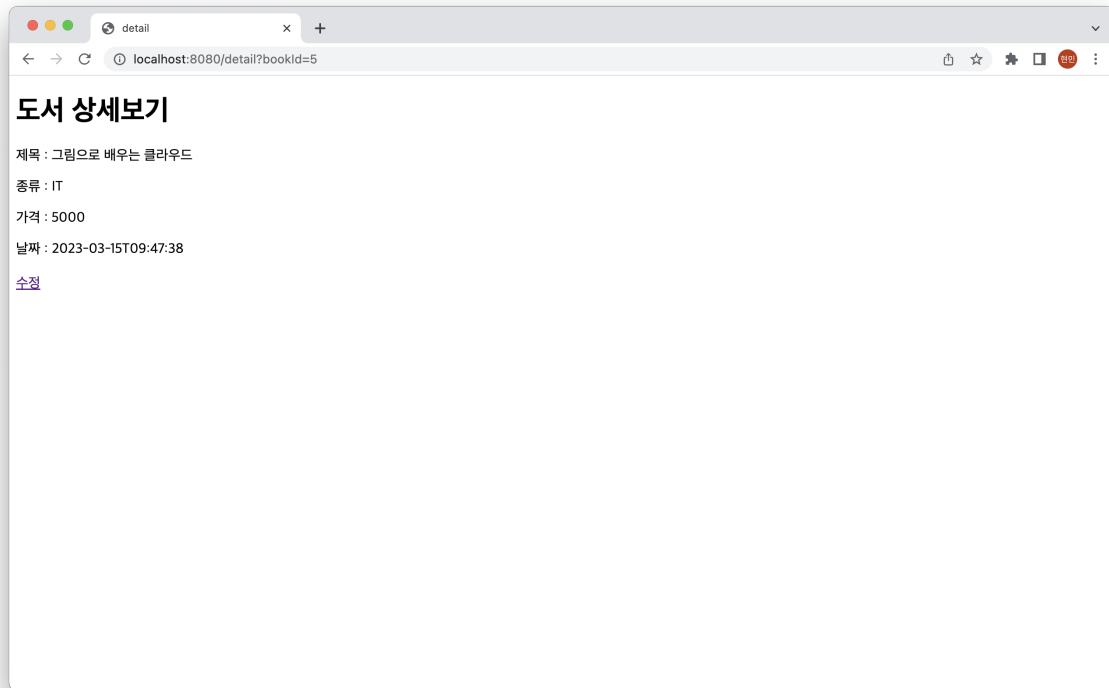
제목 :

종류 :

가격 :

MyBatis UPDATE 성공

	book_id	title	category	price	insert_date
▶	1	2	2	2	2023-03-14 12:52:30
	2	1	1	1	2023-03-14 14:07:46
	3	스프링 5 마스터	IT 서적	40000	2023-03-14 17:04:06
	4	Do it! 파이썬 생활프로그래밍	IT 서적	20000	2023-03-15 09:43:17
	5	그림으로 배우는 클라우드	IT	5000	2023-03-15 09:47:38
	NULL	NULL	NULL	NULL	NULL



BookStore SELECT All, DELETE by MyBatis

| MyBatis 이용하여 **C R U D** 처리

book_SQL.xml

```
<select id="select_list" parameterType="hashMap" resultType="hashMap">
  <![CDATA[
    SELECT
      book_id, title, category, price, insert_date
    FROM book
    ORDER BY book_id desc
  ]]>
</select>
<delete id="delete" parameterType="hashMap">
  <![CDATA[
    DELETE book
    WHERE
      book_id = #{bookId}
  ]]>
</delete>
```

Class

BookRepository

```
List<Map<String, Object>> selectList(Map<String, Object> map);

int delete(Map<String, Object> map);
```

BookRepositoryImpl

```
@Override
public List<Map<String, Object>> selectList(Map<String, Object> map) {

    return this.sqlSessionTemplate.selectList("book.select_list", map);
}

@Override
public int delete(Map<String, Object> map) {

    return this.sqlSessionTemplate.delete("book.delete", map);
}
```

BookService

```
List<Map<String, Object>> list(Map<String, Object> map);

boolean delete(Map<String, Object> map);
```

BookServiceImpl

```
@Override
public List<Map<String, Object>> list(Map<String, Object> map) {

    return this.bookRepository.selectList(map);
}

@Override
public boolean delete(Map<String, Object> map) {

    int result = this.bookRepository.delete(map);

    return result == 1;
}
```

BookController


```

@RequestMapping("/list")
public ModelAndView list(@RequestParam Map<String, Object> map) {
    List<Map<String, Object>> list = this.bookService.list(map);

    ModelAndView mav = new ModelAndView();
    mav.addObject("data", list);
    mav.setViewName("book/list");

    return mav;
}

@PostMapping("/delete")
public ModelAndView deletePost(@RequestParam Map<String, Object> map) {
    ModelAndView mav = new ModelAndView();

    boolean isDeleteSuccess = this.bookService.delete(map);
    if(isDeleteSuccess) {
        mav.setViewName("redirect:/list");
    } else {
        String bookId = map.get("bookId").toString();
        mav.setViewName("redirect:/detail?bookId=" + bookId);
    }
    return mav;
}

```

View

list.jsp

```

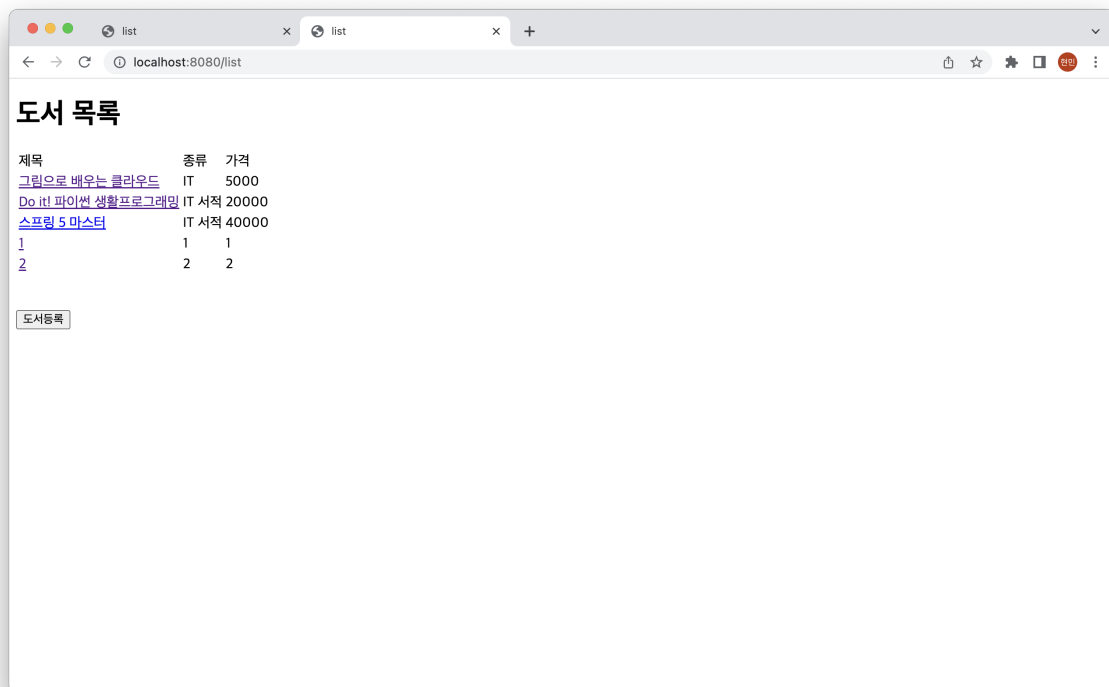
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>list</title>
</head>
<body>
    <h1>도서 목록</h1>
    <table>
        <thead>
            <tr>
                <td>제목</td>
                <td>종류</td>
                <td>가격</td>
            </tr>
        </thead>
        <tbody>
            <c:forEach var="row" items="${data }">
                <tr>
                    <td><a href="/detail?bookId=${row.book_id }">${row.title }</a>

```

```

        </td>
        <td>${row.category }</td>
        <td>${row.price }</td>
    </tr>
</c:forEach>
</tbody>
</table>
<br>
<br>
<button onclick="location.href='/create'">도서등록</button>
</body>
</html>

```



detail.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>detail</title>
</head>
<body>
    <h1>도서 상세보기</h1>
    <p>제목 : ${data.title }
    <p>종류 : ${data.category }

```

```

<p>가격 : ${data.price }
<p>날짜 : ${data.insert_date }
<br>
<br>
<button onclick="location.href='/update?bookId=${bookId}'">수정</button>
<button onclick="location.href='/list'">목록</button>
<form method="POST" action="/delete">
  <input type="hidden" name="bookId" value="${bookId }"/>
  <input type="submit" value="삭제" />
</form>
</body>
</html>

```

MyBatis DELETE 성공

	book_id	title	category	price	insert_date
▶	3	스프링 5 마스터	IT 서적	40000	2023-03-14 17:04:06
	4	Do it! 파이썬 생활프로그래밍	IT 서적	20000	2023-03-15 09:43:17
	5	그림으로 배우는 클라우드	IT	5000	2023-03-15 09:47:38
	NULL	NULL	NULL	NULL	NULL

BookStore 검색 by MyBatis

MyBatis 이용하여 검색 처리

book_SQL.xml

where 1=1

- 관습적인 표현이다. 1 과 1은 같다는 항상 참이다. 따라서 검색 조건을 and로 연결하기 위해 사용된다.

<if>...</if>

- 마이바티스에서 조건을 표현한다. test 는 조건의 규칙을 나타내는 표현이다.
- 만약 키워드가 있으면 (null이 아니고 값이 없지도 않으면) if 구문이 쿼리에 포함된다.
- 이와 같이 쿼리의 내용이 파라미터가 아닌 마이 바티스의 규칙에 따라 변경되는 것을 동적 쿼리 Dynamic Query 라고 한다.

like '이%'	'이' 로 시작	이사, 이동
like '%이'	'이' 로 끝	갑쟁이, 일찍이
like '%이%'	앞뒤 상관없이 '이' 포함	이미지, 파이썬

```

<select id="select_list" parameterType="hashMap"
    resultType="hashMap">
    <![CDATA[
        SELECT
        book_id, title, category, price, insert_date
        FROM book
        WHERE 1=1
    ]]>
    <if test="keyword != null and keyword != ''">
        and (title like CONCAT('%',{keyword},%') or category like CONCAT('%',{keyword},%'))
    </if>
    order by insert_date desc
</select>

```

Class

BookController

```

@RequestMapping(value="/list")
public ModelAndView list(@RequestParam Map<String, Object> map) {

    List<Map<String, Object>> list = this.bookService.list(map);

    ModelAndView mav = new ModelAndView();
    mav.addObject("data", list);

    if (map.containsKey("keyword")) {
        mav.addObject("keyword", map.get("keyword"));
    }

    mav.setViewName("/book/list");
    return mav;
}

```

View

list.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>list</title>
</head>
<body>

```

```

<h1>도서 목록</h1>
<p>
<form>
  <input type="text" placeholder="검색" name="keyword" value="${keyword}" />
  <input type="submit" value="검색" />
</form>
</p>
<table>
  <thead>
    <tr>
      <td>제목</td>
      <td>종류</td>
      <td>가격</td>
    </tr>
  </thead>
  <tbody>
    <c:forEach var="row" items="${data }">
      <tr>
        <td><a href="/detail?bookId=${row.book_id }">${row.title }</a>
        </td>
        <td>${row.category }</td>
        <td>${row.price }</td>
      </tr>
    </c:forEach>
  </tbody>
</table>
<br>
<br>
<button onclick="location.href='/create'">도서등록</button>
</body>
</html>

```

