Mybatis



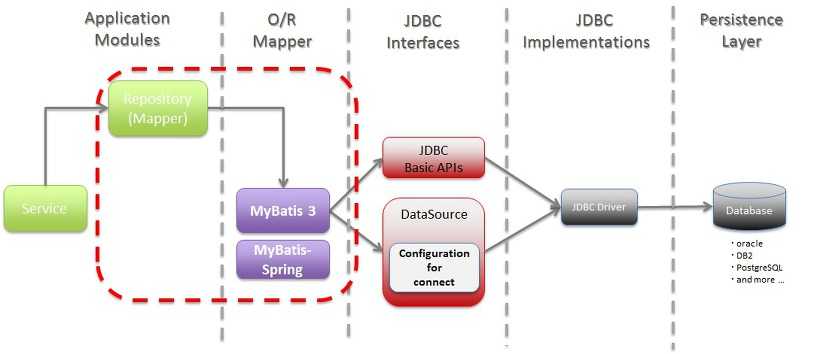
# MyBatis 란..??

* Java Object와 SQL문 사이에서 자동 Mapping 기능을 지원하는 ORM(Object Relational Mapping) Framework

# MyBatis의 특징

* JDBC를 통해 RDBMS(관계형 데이터베이스 관리 시스템)에 엑세스하는 작업을 캡슐화하고 기존 JDBC의 중복작업을 간소화
* XML파일의 형태인 mapper를 통해 프로그램 코드로 부터 SQL 쿼리를 분리
* 쿼리 실행 결과를 Java 객체와 자동으로 매핑
* 빠른 개발과 생산성의 향상
* 기존 JDBC보다 사용하기 편리
* 학습의 부담이 적음

# Mybatis의 아키텍처



기존 JDBC 프로그래밍의 경우 Repository(DAO)에서 곧바로 JDBC API쪽으로 접근하여 DB를 연결하였지만, 위의 그림에 나와있듯이 Mybatis을 사용할 경우 Repository와 JDBC API사이에 MyBatis가 위치함으로써 편리한 Access를 제공한다.

# Mybatis의 Mapper(매퍼)

DAO클래스를 사용하면

* SQL문만 수정하더라도 클래스를 다시 컴파일 해야한다.
* SQL문만 별도로 한 곳에 모아서 관리하기 어렵다.

마이바티스는 Mapper를 통해 SQL문을 Java코드로 부터 완전히 분리한다.

따라서 SQL 문에 대한 통합 관리를 할 수 있다.

매퍼는 xml 문서이므로 다음과 같은 포맷을 가진다.



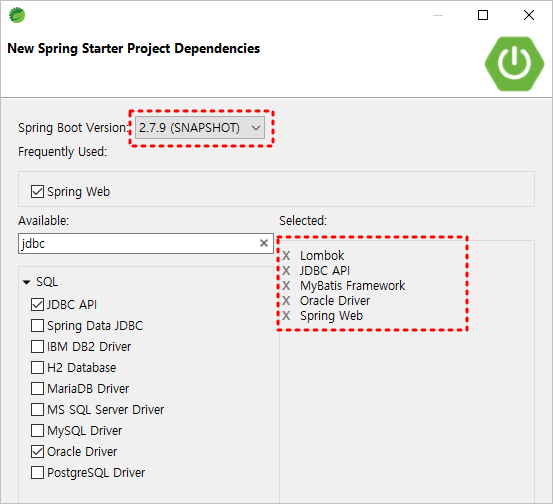
# Mapper 의 Doctype

| <!DOCTYPE mapper  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"  "http://mybatis.org/dtd/mybatis-3-mapper.dtd"> |
| --- |

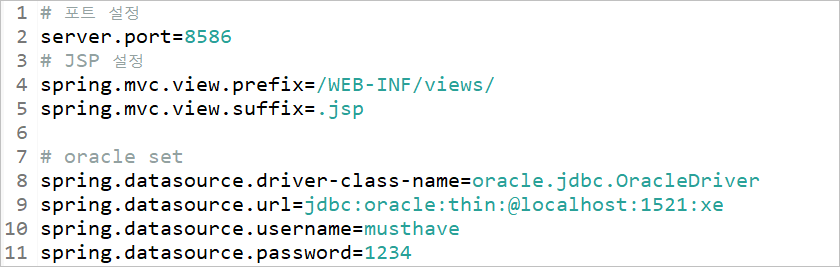
# 프로젝트명 : B07aMybatis

## 준비사항

* 의존설정 : Spring Web, Lombok, JDBC API, Oracle Driver, Mybatis Framework
* JSP 사용을 위한 설정을 한다.
* Refresh Gradle Project 를 눌러 적용한다.

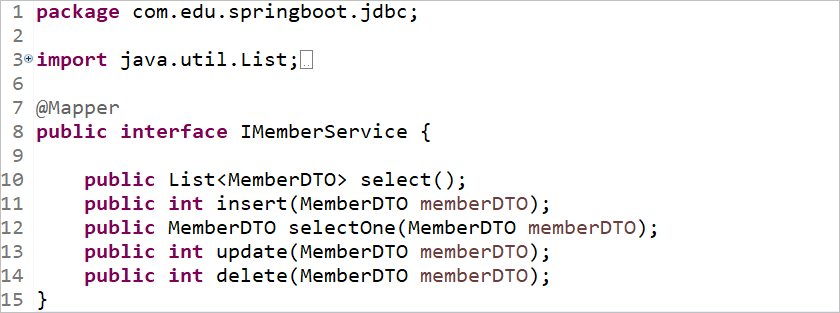


### 프로퍼티 : resources/application.properties

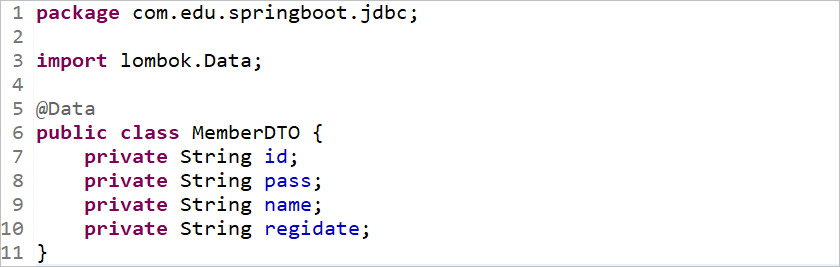




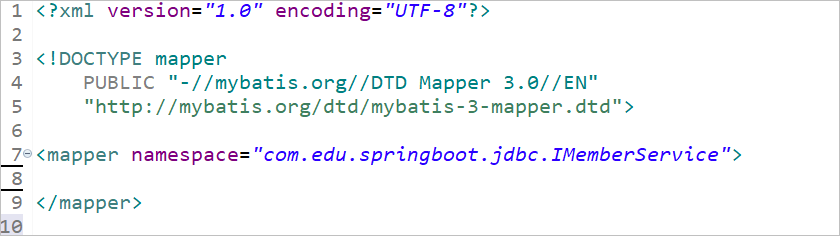
### 서비스 인터페이스 : com.edu.springboot.jdbc.IMemberService.java



### DTO : com.edu.springboot.jdbc.MemberDTO.java



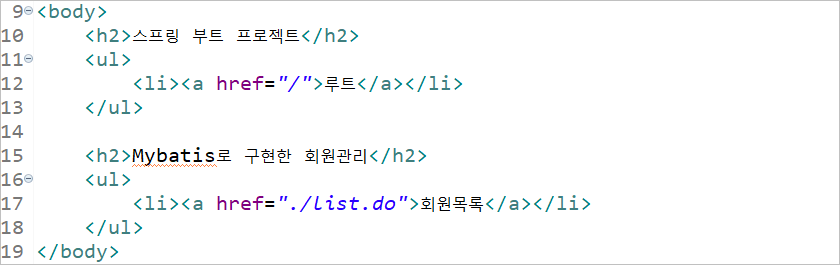
### 매퍼 : resources/mybatis/mapper/MemberDAO.xml [위 설명부분 참조]



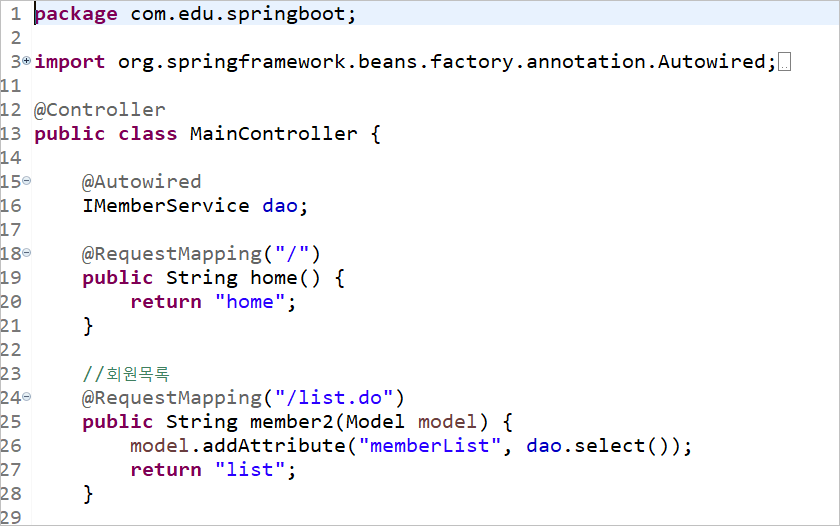
**여기까지 작성하세요.**

## 회원목록

### home : webapp/WEB-INF/views/home.jsp

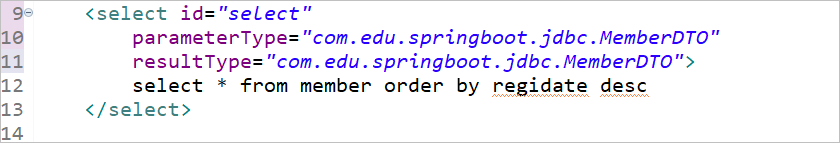


### 컨트롤러 : com.edu.springboot.MainController.java

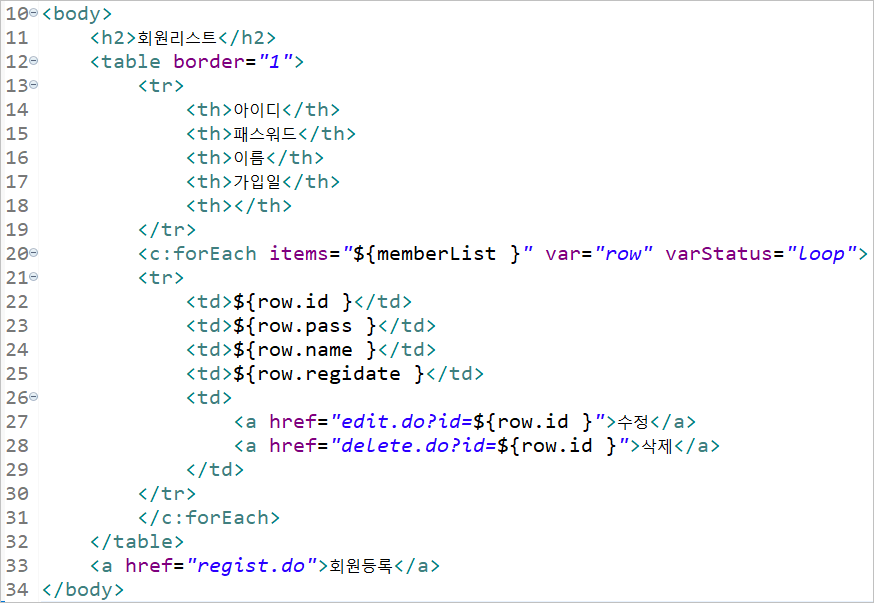


**여기까지 작성하세요.**

### 매퍼 : resources/mybatis/mapper/MemberDAO.xml



### 뷰 : webapp/WEB-INF/views/list.jsp

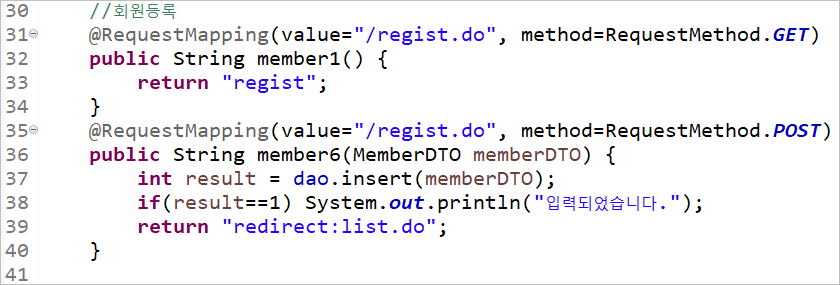


여기까지 작성하세요.

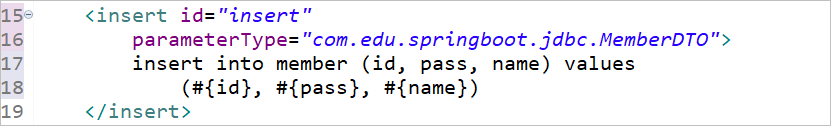
## 

## 등록하기

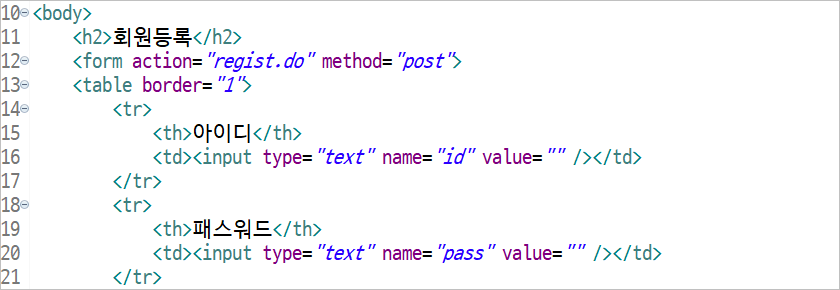
### 컨트롤러 : com.edu.springboot.MainController.java

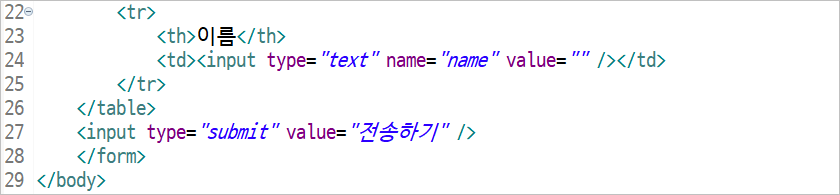


### 매퍼 : resources/mybatis/mapper/MemberDAO.xml



### 뷰 : webapp/WEB-INF/views/regist.jsp



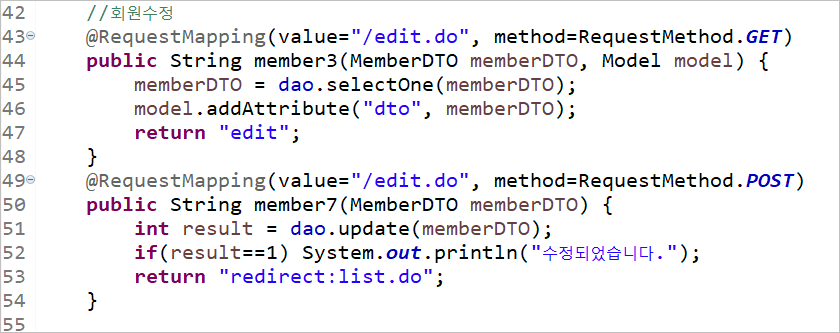


여기까지 작성하세요.

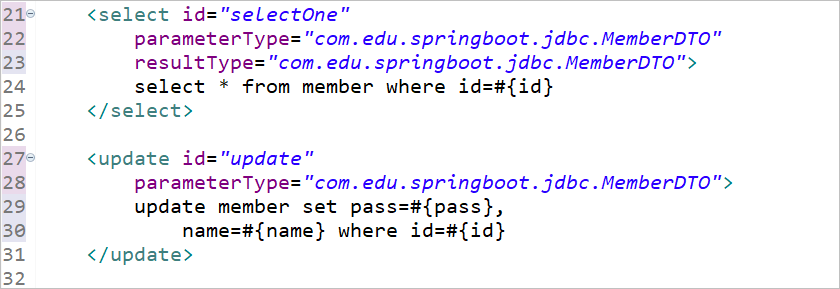
## 

## 수정하기

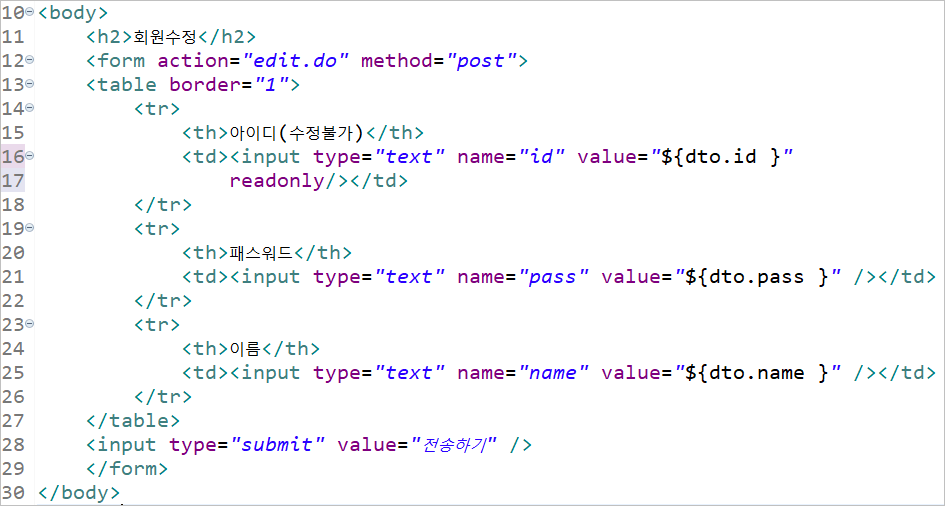
### 컨트롤러 : com.edu.springboot.MainController.java



### 매퍼 : resources/mybatis/mapper/MemberDAO.xml



### 뷰 : webapp/WEB-INF/views/edit.jsp



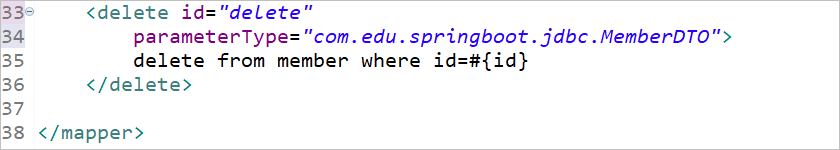
여기까지 작성하세요.

## 

## 삭제하기

### 컨트롤러 : com.edu.springboot.MainController.java

### 매퍼 : resources/mybatis/mapper/MemberDAO.xml



# 퀴즈]

현재 삭제 버튼을 누르는 경우 즉시 삭제된다. Javascript의 confirm() 함수로 물어본 후 '확인'을 누를때만 삭제될 수 있도록 구현하시오. 단, 전송방식은 post로 처리한다.

