Spring MyBatis 연동

DB 연결정보는 애플리케이션이 동작하는 환경에 따라 자주 바뀔 수 있음

=> 자주 정리되는 값들을 properties 파일로 분리

키와 값의 쌍(key=value , ex. jdbc.username=ezen)으로 설정

프로퍼티 파일로 분리한 정보를 ${}(프로퍼티 치환자)을 이용하여 설정함

xml파일 설정

pom.xml에 라이브러리 추가

dependency

mybatis, mybatis-spring : 스프링과 마이바티스 연동

ojdbc, spring-jdbc : 스프링에서 데이터베이스 처리와 트랜잭션 처리

ojdbc: 오라클에서 제공하는 오라클 DB와 자바 애플리케이션을

연결하기 위한 라이브러리 파일

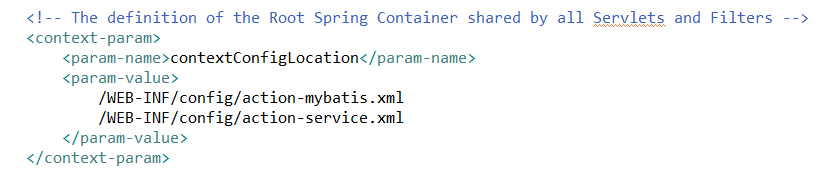
<!-- https://mvnrepository.com/artifact/org.mybatis/mybatis →

<!-- https://mvnrepository.com/artifact/org.mybatis/mybatis-spring →

<!-- https://mvnrepository.com/artifact/com.oracle.database.jdbc/ojdbc8 -->

<!-- https://mvnrepository.com/artifact/org.springframework/spring-jdbc -->

web.xml 설정



<context-param>

<init param>과 달리 여러 서블릿의 공통의 환경정보를 제공,

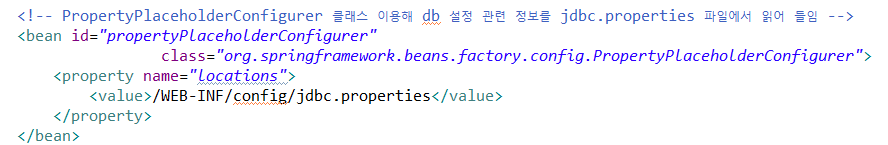
웹 어플리케이션의 서블릿들이 공유하는 매개변수

web.xml에 바로 선언한다

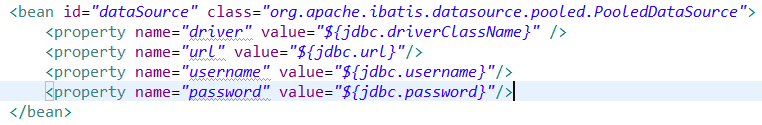
어플리케이션 실행 시 spring-myBatis 연결 설정한 config폴더 내 xml들을

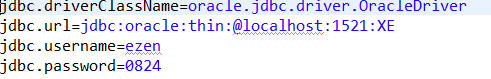
불러옴

config/action-mybatis.xml설정



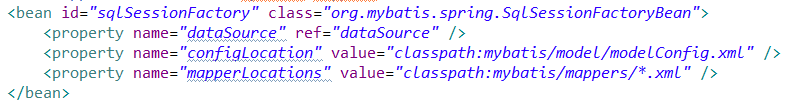
.PropertyPlaceholderConfigurer 이용해 jdbc.properties파일의 값들을 읽어들임





PooledDataSource 이용해서 dataSource 빈 생성하고 jdbc.properties의 값들을 주입

<SqlSessionFactoryBean>설정



SqlSessionFactoryBean을 이용해 SqlSessionFactory를 생성

<property>로 설정해야 할 것들

name = “dataSource” ref = “dataSource”

연결을 위한 dataSource

name="configLocation

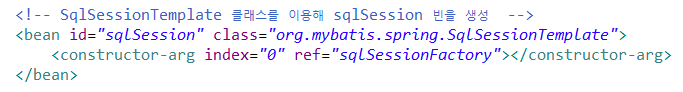
"value="classpath:mybatis/model/modelConfig.xml"

쿼리문의 결과값을 DAO로 전달하기 위한 bean(memberDTO)

value에 경로값을 넣기 위해서 classpath: 사용

name="mapperLocations" value="classpath:mybatis/mappers/\*.xml"

쿼리문이 있는

<SqlSessionFactoryTemplate>

SqlSessionFactoryTemplate을 이용해 SqlSession 빈 생성

생성한 SqlSessionFactory 주입