

Spring Framework

16. MyBatis 개요

CONTENTS

1

MyBatis의 개요와 특징

2

MyBatis와 MyBatis-Spring의 주요 컴포넌트

3

MyBatis-Spring을 사용한 예제

학습 목표

- MyBatis 개요와 특징에 대하여 이해할 수 있습니다.
- MyBatis와 MyBatis-Spring의 주요 컴포넌트에 대하여 이해할 수 있습니다.
- MyBatis-Spring을 사용한 예제에 대하여 이해할 수 있습니다.



1. MyBatis 개요와 특징

■ MyBatis의 개요

MyBatis(<http://www.mybatis.org/mybatis-3/>)는
자바 오브젝트와 SQL문 사이의 자동 Mapping 기능을 지원하는
ORM 프레임워크이다.

- MyBatis는 SQL을 별도의 파일로 분리해서 관리하게 해주며,
객체-SQL 사이의 파라미터 Mapping 작업을 자동으로 해주기 때문에
많은 인기를 얻고 있는 기술이다.
- MyBatis는 Hibernate나 JPA(Java Persistence Api)처럼 새로운 DB
프로그래밍 패러다임을 익혀야 하는 부담이 없이, 개발자가 익숙한
SQL을 그대로 이용하면서 JDBC 코드 작성의 불편함도 제거해주고,
도메인 객체나 VO 객체를 중심으로 개발이 가능하다는 장점이 있다.

■ MyBatis의 특징

01 쉬운 접근성과 코드의 간결함

- ◉ 가장 간단한 퍼시스턴스 프레임워크
- ◉ XML 형태로 서술된 JDBC 코드라고 생각해도 될 만큼 JDBC의 모든 기능을 MyBatis가 대부분 제공한다.
- ◉ 복잡한 JDBC코드를 걷어내며 깔끔한 소스코드를 유지할 수 있다.
- ◉ 수동적인 파라미터 설정과 쿼리 결과에 대한 맵핑 구문을 제거할 수 있다.

■ MyBatis의 특징

02 SQL문과 프로그래밍 코드의 분리

- ◉ SQL에 변경이 있을 때마다 자바 코드를 수정하거나 컴파일 하지 않아도 된다.
- ◉ SQL 작성과 관리 또는 검토를 DBA와 같은 개발자가 아닌 다른 사람에게 맡길 수도 있다.

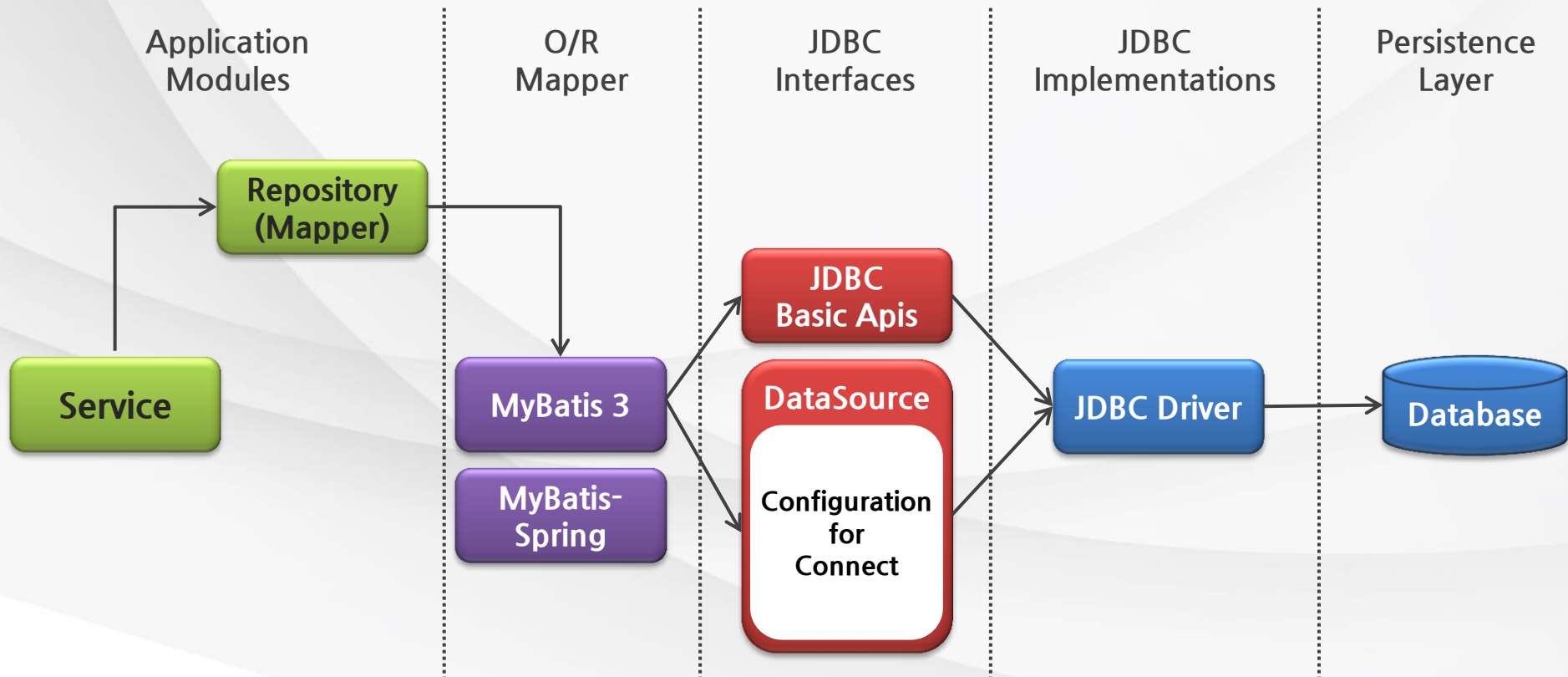
03 다양한 프로그래밍 언어로 구현가능

- ◉ Java, C#, .NET, Ruby

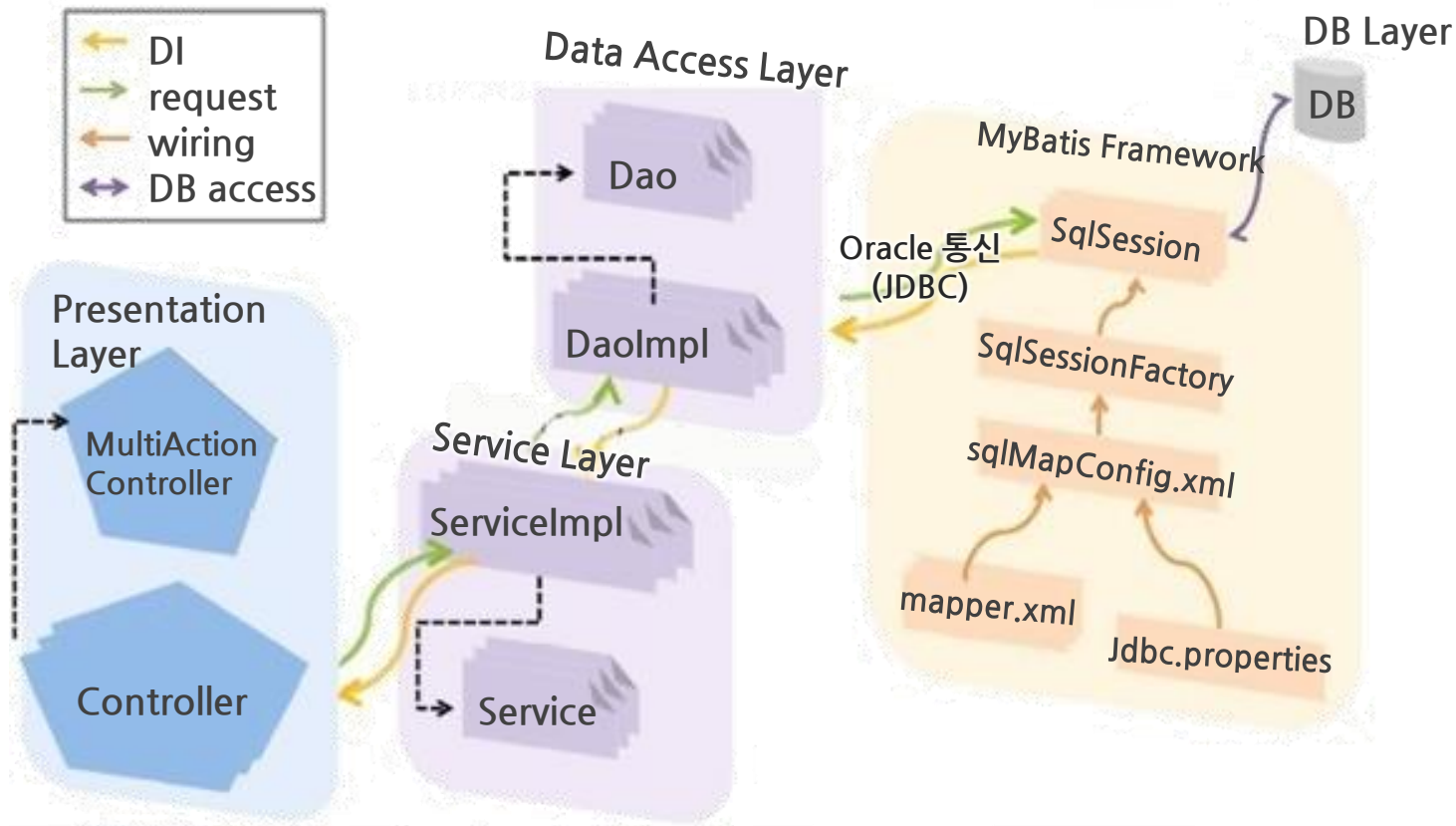
A person's hands are shown holding a smartphone, with the screen glowing. The background is dark with out-of-focus, colorful bokeh lights in shades of yellow, orange, and blue. A semi-transparent dark banner with a yellow decorative element on the left contains the title text.

2. MyBatis3와 MyBatis-Spring의 주요 컴포넌트

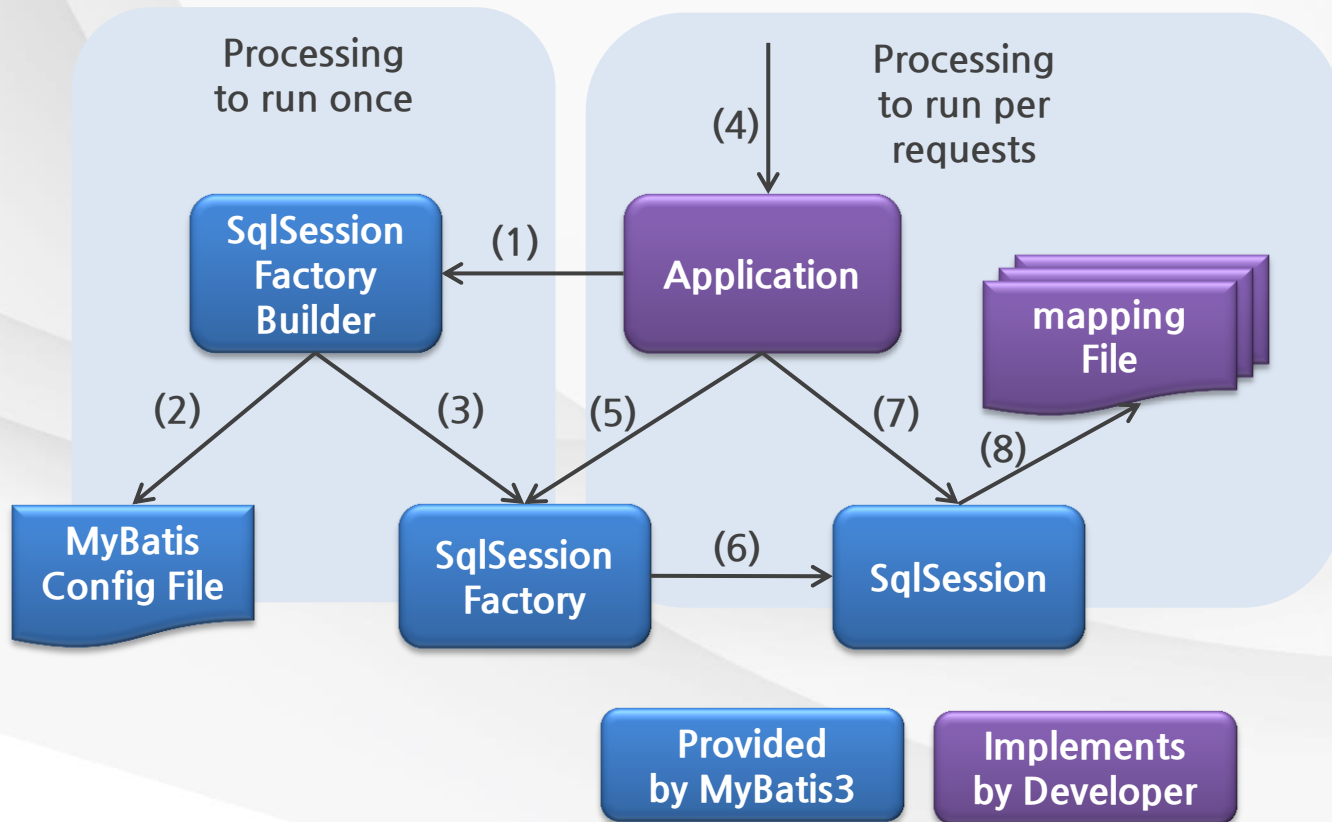
■ MyBatis와 MyBatis-Spring을 사용한 DB 액세스 Architecture



■ MyBatis를 사용하는 데이터 액세스 계층



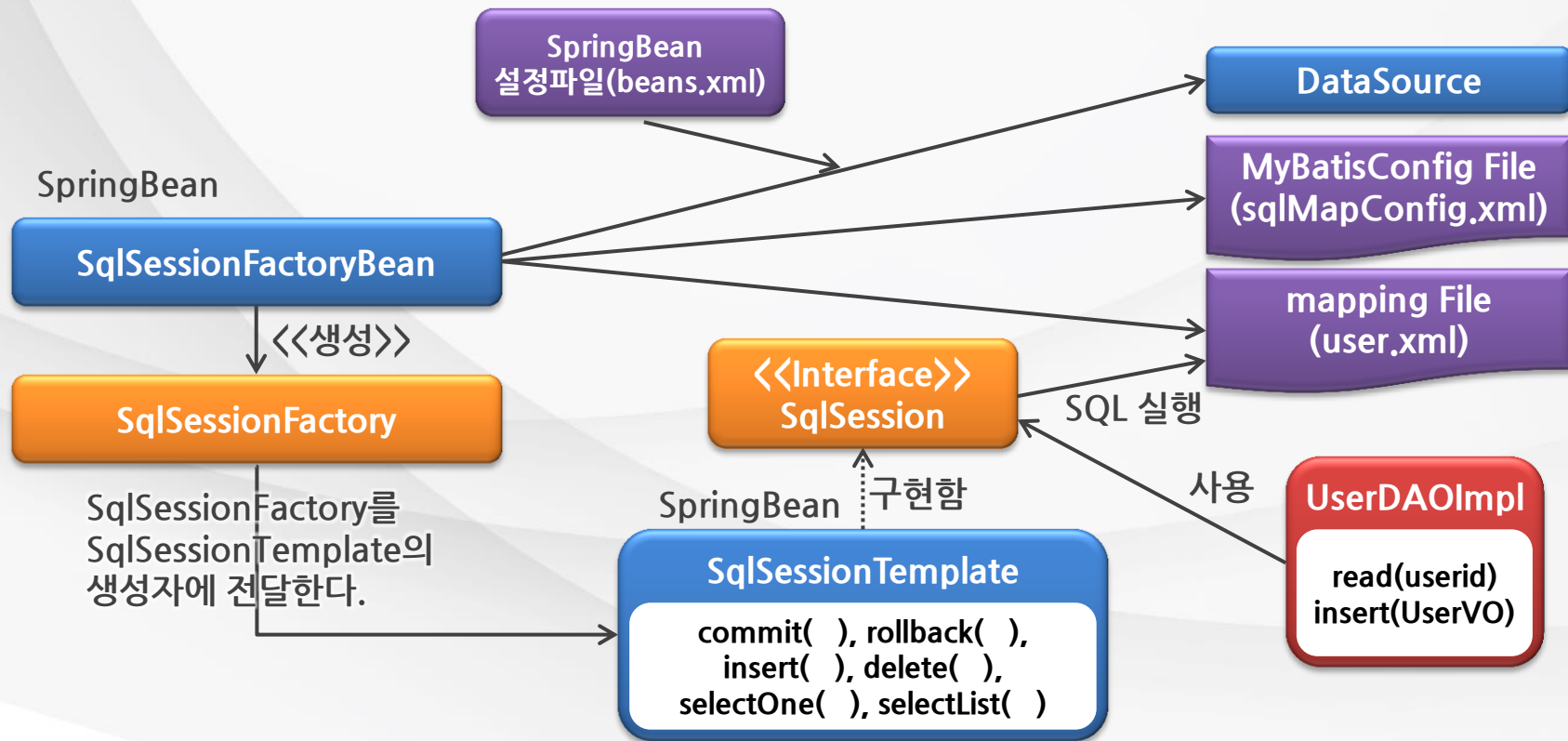
■ MyBatis3의 주요 컴포넌트



■ MyBatis3의 주요 컴포넌트의 역할

MyBatis 설정파일 (SqlMapConfig.xml)	▪ 데이터베이스의 접속 주소 정보나 Mapping 파일의 경로 등의 고정된 환경정보를 설정한다.
SqlSessionFactoryBuilder	▪ MyBatis 설정 파일을 바탕으로 SqlSessionFactory를 생성한다.
SqlSessionFactory	▪ SqlSession을 생성한다.
SqlSession	<ul style="list-style-type: none">▪ 핵심적인 역할을 하는 클래스로서 SQL 실행이나 트랜잭션 관리를 실행한다.▪ SqlSession 오브젝트는 Thread-Safe 하지 않으므로 thread마다 필요에 따라 생성한다.
mapping 파일 (user.xml)	▪ SQL문과 OR Mapping을 설정한다.

■ MyBatis-Spring의 주요 컴포넌트



■ MyBatis-Spring의 주요 컴포넌트의 역할

MyBatis 설정파일
(sqlMapConfig.xml)

- VO 객체의 정보를 설정한다.

SqlSessionFactoryBean

- MyBatis 설정파일을 바탕으로 SqlSessionFactory를 생성한다.
- Spring Bean으로 등록해야 한다.

SqlSessionTemplate

- 핵심적인 역할을 하는 클래스로서 SQL 실행이나 트랜잭션 관리를 실행한다.
- SqlSession 인터페이스를 구현하며, Thread-safe하다.
- Spring Bean으로 등록해야 한다.

■ MyBatis-Spring 의 주요 컴포넌트의 역할

Mapping 파일 (user.xml)

- SQL문과 OR Mapping을 설정한다.

Spring Bean 설정파일(beans.xml)

- SqlSessionFactoryBean을 Bean 등록할 때 DataSource 정보와 MyBatis Config 파일정보, Mapping 파일의 정보를 함께 설정한다.
- SqlSesseionTemplate을 Bean으로 등록한다.

A person's hands are shown holding a smartphone, with the screen glowing. The background is dark with out-of-focus, colorful bokeh lights in shades of yellow, orange, and blue. A semi-transparent dark banner is at the bottom, containing a yellow decorative mark and the section title.

3. MyBatis-Spring을 사용한 예제

■ Mapping 파일 (SQL문 포함)

```
*User.xml
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE mapper PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"
3   "http://mybatis.org/dtd/mybatis-3-mapper.dtd">
4 <mapper namespace="userNS">
5
6   <select id="selectUserById" parameterType="string" resultType="User">
7     select * from users where userid=#{value}
8   </select>
9   <insert id="insertUser" parameterType="User">
10    insert into users values(#{userId},#{name},#{gender},#{city} )
11  </insert>
```

■ MyBatis Configuration 파일

SqlMapConfig.xml

```
1 <?xml version="1.0" encoding="UTF-8" ?>
2 <!DOCTYPE configuration PUBLIC "-//mybatis.org//DTD Config 3.0//EN"
3   "http://mybatis.org/dtd/mybatis-3-config.dtd">
4
5 <configuration>
6   <typeAliases>
7     <typeAlias alias="User" type="my.spring.vo.UserVO" />
8     <typeAlias alias="Student" type="my.spring.vo.Student" />
9     <typeAlias alias="Dept" type="my.spring.vo.Dept" />
10  </typeAliases>
11 </configuration>
```

■ Spring Bean 설정 파일

```
beans.xml
126
127 <!-- MyBatis 설정 -->
128 <bean id="sqlSessionFactory" class="org.mybatis.spring.SqlSessionFactoryBean">
129     <property name="dataSource" ref="dataSource" />
130     <property name="configLocation" value="classpath:config/SqlMapConfig.xml" />
131     <property name="mapperLocations">
132         <list>
133             <value>classpath:config/User.xml</value>
134             <value>classpath:config/Student.xml</value>
135         </list>
136     </property>
137 </bean>
138 <!-- SqlSessionTemplate 설정 -->
139 <bean id="sqlSession" class="org.mybatis.spring.SqlSessionTemplate">
140     <constructor-arg ref="sqlSessionFactory" />
141 </bean>
```

■ 데이터 액세스 계층의 DAO 구현 클래스

UserDaoImpl.java

```
11 @Repository("userDao")
12 public class UserDaoImpl implements UserDao {
13     @Autowired
14     private SqlSession session;
15
16     public void insert(UserVO user) {
17         session.update("userNS.insertUser", user);
18         System.out.println("등록된 Record UserId=" + user.getUserId() + " Name=" + user.getName());
19     }
20     @Override
21     public UserVO read(String id) {
22         UserVO user = session.selectOne("userNS.selectUserById", id);
23         return user;
24     }
}
```

User.xml →

```
<mapper namespace="userNS">
    <select id="selectUserById" parameterType="string" resultType="User">
        select * from users where userid=#{value}
    </select>
    <insert id="insertUser" parameterType="User">
        insert into users values(#{userId},#{name},#{gender},#{city} )
    </insert>
</mapper>
```



학습정리

지금까지 **[MyBatis 개요]**에 대해서 살펴보았습니다.

MyBatis 개요와 특징

MyBatis는 ORM 프레임워크, 자바 코드와 SQL를 분리함

MyBatis 와 MyBatis-Spring 주요 컴포넌트

- ◉ SqlSessionFactoryBuilder, SqlSessionFactory, SqlSession
- ◉ SqlSessionFactoryBean, SqlSessionTemplate

MyBatis-Spring을 사용한 예제

MyBatis Config 파일, Mapping 파일 , Spring Bean 설정 파일
DAO 클래스에서 SqlSession의 사용