

마이바티스

# MyBatis의 특징

- SQL문의 일반적인 실행 방법
  - SQL 문을 문자열입력 => 실행
  - SQL문의 복잡성이 늘어남에 따라 유지보수가 어려워짐
- 마이바티스 프레임워크는 자바코드와 SQL문을 분리해서 가독성을 높이고 사용 및 유지 보수를 편리하게 함

# MyBatis의 특징

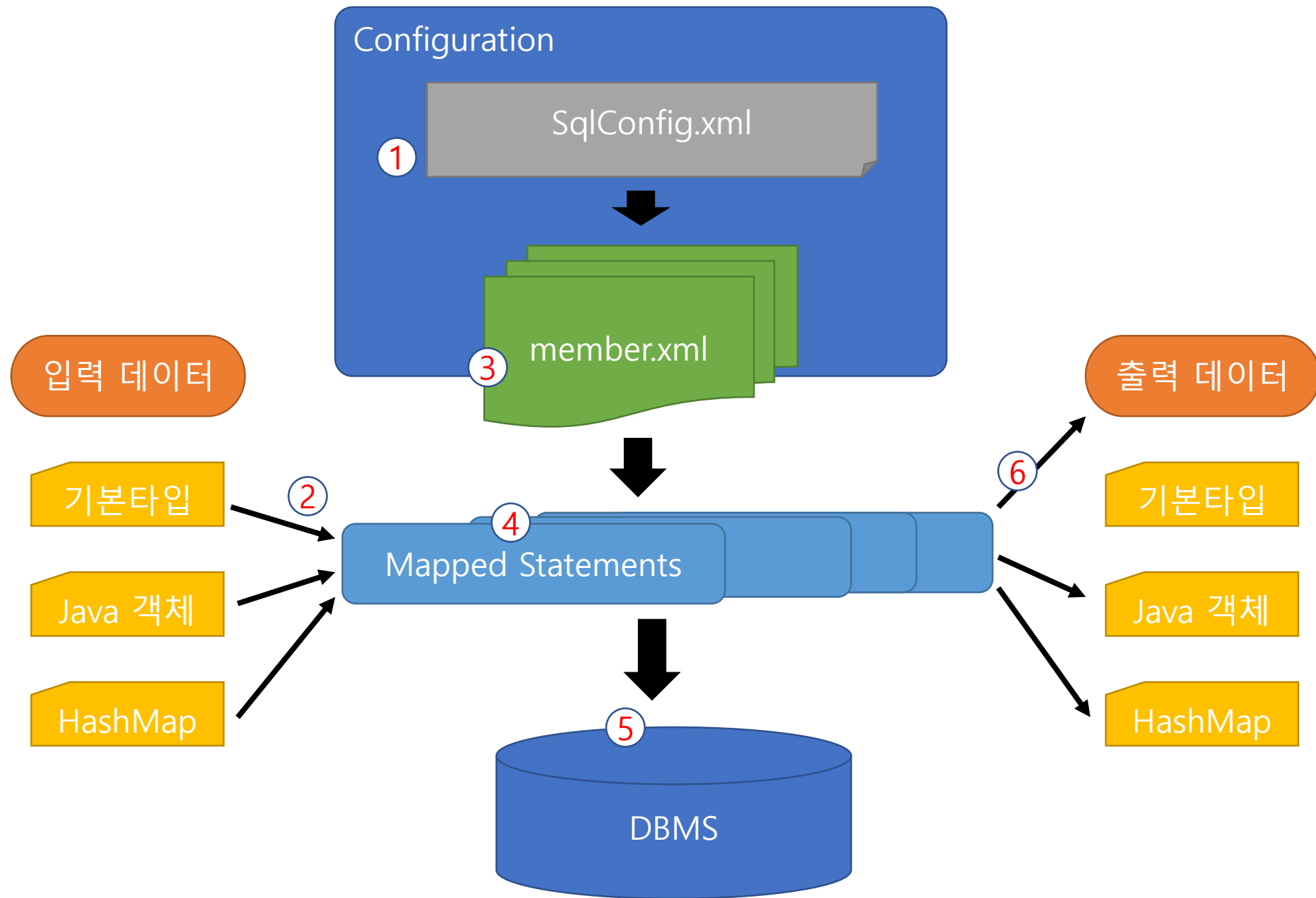
- 특징

- SQL 실행 결과를 자바빈즈 또는 Map 객체에 매핑해주는 Persistence 솔루션으로 관리함(즉, SQL을 소스코드가 아닌 XML 로 분리)
- SQL과 프로그래밍 코드를 분리해서 구현
- 데이터 소스기능과 트랜잭션 처리 기능을 제공

# MyBatis의 특징

- 절차

1. SqlMapConfig.xml에 각 기능별 실행할 SQL문을 sqlMap.xml에 미리 작성한 후 등록
2. 애플리케이션에서 데이터베이스와 연동하는 데 필요한 데이터를 각각의 매개변수에 저장한 후 마이바티스에 전달
3. 애플리케이션에서 요청한 SQL 문을 SqlMap.xml에서 선택
4. 전달 한 매개변수와 선택한 SQL 결합
5. 매개변수와 결합된 SQL문을 DBMS에서 실행
6. DBMS에서 반환된 데이터를 애플리케이션에서 제공하는 적당한 매개변수에 저장한 후 반환



# MyBatis의 예제

- 기본적인 서블릿/JSP 프로그램에서 MyBatis를 적용 해 봅시다.
- 우선 MyBatis가 적용되지 않은 코드부터 만들어 봅니다.

- com.green.vo
  - MemberVo.java
- com.green.db
  - DBManager.java
- com.green.dao
  - MemberDAO.java
- com.green.controller
  - FirstServlet.java

# MyBatis의 예제

- 앞의 예제를 MyBatis코드로 전환하려면
- 만들어진 설정파일을 읽어와서 SqlSessionFactory를 빌드해야 한다.
- 빌드된 SqlSessionFactory객체를 통해 SqlSession객체를 얻어오고 이를 통해 Mapper에 미리 작성된 쿼리를 실행함으로 데이터베이스에 접근한다.

# MyBatis의 예제

- 기존 코드를 myBatis로 변환한다.

- MyBatis.config
  - SqlConfig.xml
- MyBatis.mapper
  - memberMapper.xml
- Com.green.dao
  - MemberDao2.java
- Com.green.controller
  - FirstServlet.java 수정



# MyBatis의 예제

- 전체 회원 수를 반환하는 코드를 작성해 본다(반환 타입 : 정수)
- 이메일을 통해서 회원 정보를 읽어오는 코드를 작성해 본다.(매개값 : 문자열)
- 각각 컨트롤러와 뷰페이지를 만들어서 테스트 해본다.

# MyBatis의 예제

- 데이터 조회 기능을 테스트해봤으니 다음은 Insert,Update,Delete기능을 테스트 해본다.
  - 기본 기능은 앞서 본 mapper와 유사하다  
=> 매개변수로 사용할 값을 전달해 주면 된다.
- \*다만 DAO를 생성시 MyBatis는 수동 커밋을 해주어야 한다.

# MyBatis의 예제

- MyBatis의 강점은 단순히 SQL쿼리와 Java코드간의 분리가 아니다.
- 기존SQL쿼리의 재사용성이 강점이다.

## 예시

### ◆ 자주 사용되는 Select 조건절

- `SELECT * FROM member`
- `SELECT * FROM member WHERE email='email';`
- `SELECT * FROM member WHERE name='name';`

공통의 SQL문에 조건값의 유무에 따라 동적으로 SQL문에 조건절 추가

# MyBatis의 예제

- 동적 쿼리 생성에 사용되는 태그요소

⇒if, choose, trim, foreach~~

## 예시

- 이메일 또는 이름을 받아서 회원 정보를 출력하는 쿼리를 작성해본다.

## 예시

- 이름 정보를 리스트에 담아서 조회하는 쿼리를 작성해본다.

# MyBatis의 예제

- 마이바티스를 스프링에 적용해보자

1. 메이븐 레포지터리로 부터 모듈 가져오기 – POM.xml
  - Mybatis
  - Mybatis-spring
2. 스프링에 마이바티스 빈 적용 - spring-mybatis.xml
  - DAO를 포함 4개의 빈 객체 생성
  - DataSource
  - SqlSessionFactory
  - SqlSession
  - MemberDao
3. 마이바티스 설정 등록
  - mybatis/conf/mybatisConfig.xml
4. Mapper 설정 등록(쿼리)
  - classpath:mybatis/mappers/\*.xml
  - Member.xml에 쿼리 등록
5. DAO에 메서드와 쿼리 연결