

JPA와 스프링부트를 이용한 웹 개발

금오공과대학교 김성렬

강의소개

• 목표

- 자바진영의 ORM(Object Relational Mapping) 표준인 JPA 기술 이해
- Spring Boot를 이용한 웹 개발 학습
- 총 22강

• 준비사항

- 웹 프로그래밍에 대한 전반적인 이해
- Java 언어
- 관계형 데이터베이스에 기초 지식

강의소개

• 내용

- JPA(Hibernate), Thymeleaf(템플릿 엔진), Spring Boot



Seed Starter List

Date planted	Covered	Type	Features	Rows		
03/20/2011	yes	Wood	Seed starters-specific substrate, PH Corrector used	1	Thymus vulgaris	10
				2	Thymus pseudolaginosus	15
				3	Thymus x citriodorus	20
03/25/2011	no	Plastic	Fertilizer used	1	Thymus herba-barona	5
				2	Thymus serpyllum	10

Add new Seed Starter

Seed Starter data

Date planted (MM/dd/yyyy)

Covered ☐

Type

Features

☐ Seed starter-specific substrate

☐ Fertilizer used

☐ PH Corrector used

Rows

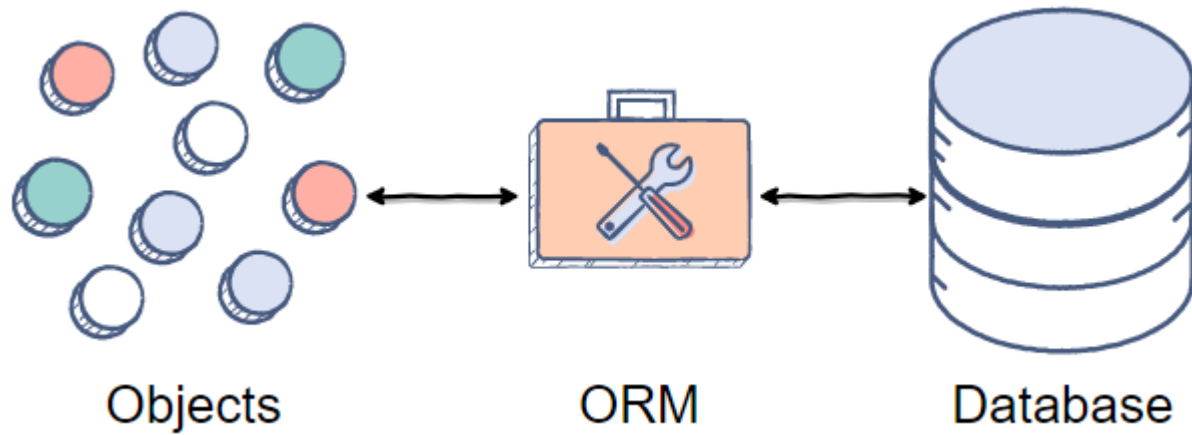
Row	Variety	Seeds per cell	
1	<input type="text" value="Thymus vulgaris"/>	<input type="text"/>	<input type="button" value="Add row"/> <input type="button" value="Remove row"/>

[출처] <https://www.thymeleaf.org/doc/tutorials/3.0/thymeleafspring.html#spring-thyme-seed-starter-manager>

ORM(Object Relational Mapping)

ORM

- ORM(Object Relational Mapping): 객체 관계 매핑



- ORM의 역할

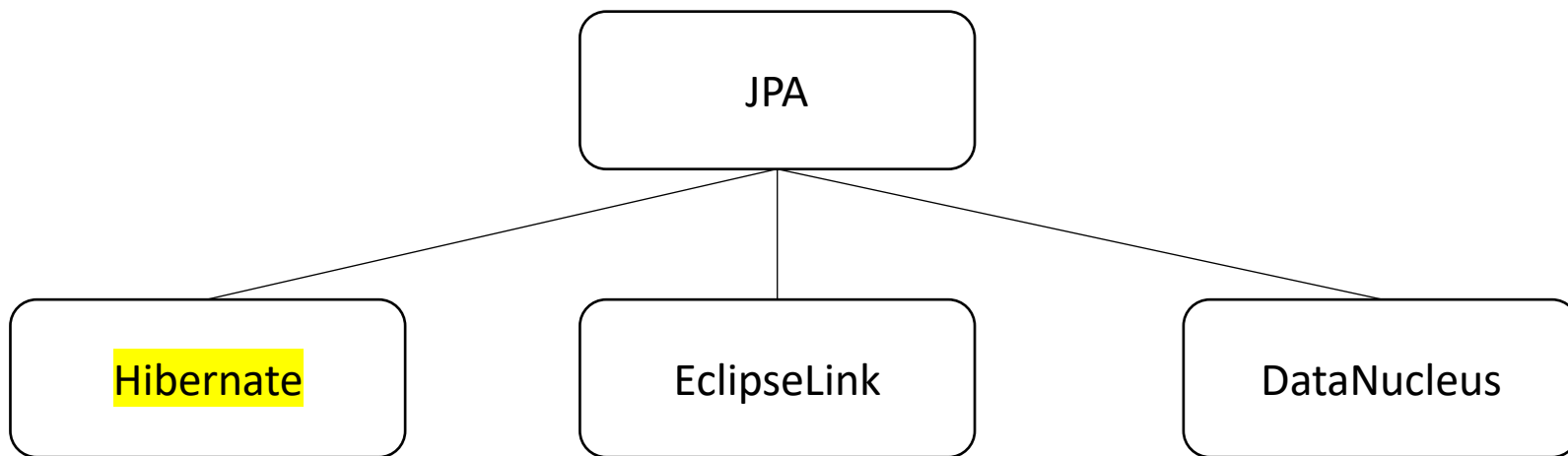
- 자바 클래스와 데이터베이스 table을 매핑

JPA

- JPA → Java Persistence API

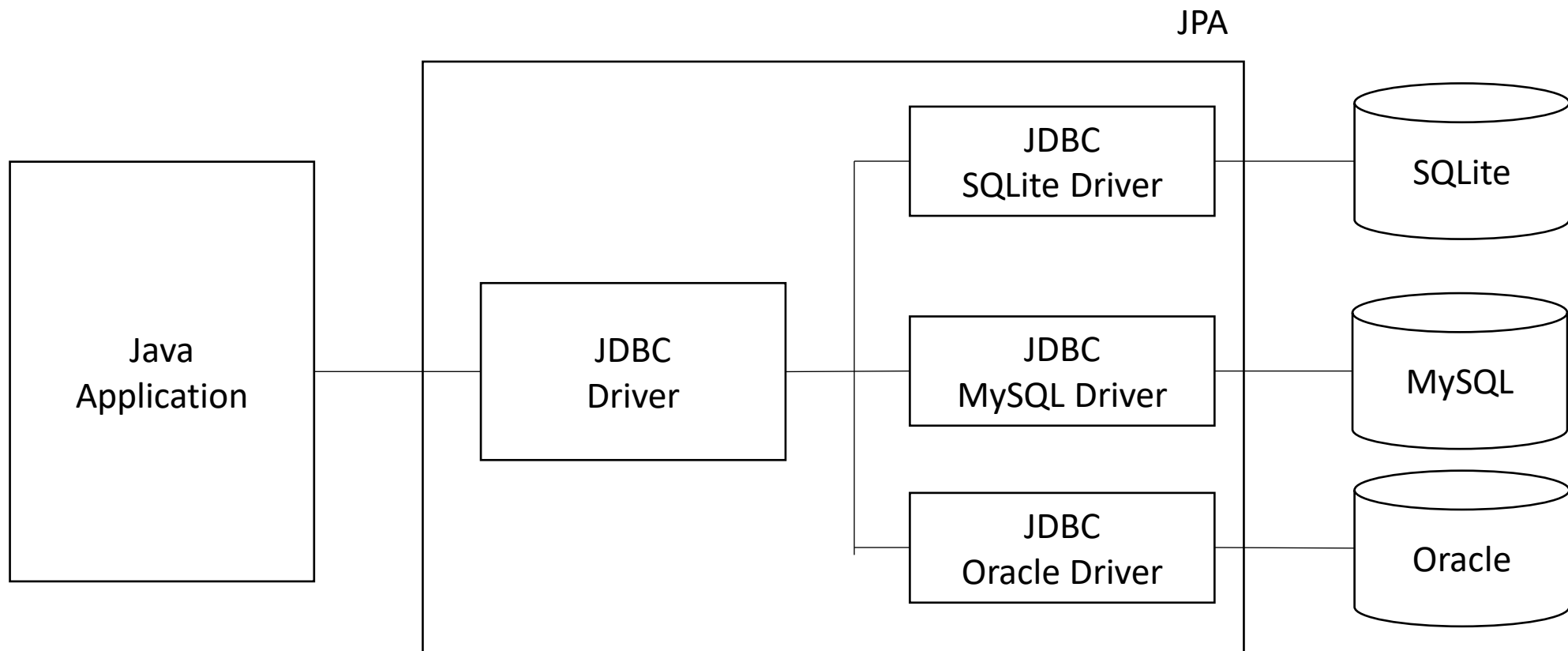
- 자바 어플리케이션에서 관계형 데이터베이스를 사용하는 방식을 정의한 [인터페이스](#)
- 자바 진영의 ORM 표준

- 인터페이스에는 구현체가 존재해야 함



JPA

- JPA도 결국 JDBC를 사용함



JPA

- JPA 사용 시 장점

- CRUD(기본 쿼리) 자동 생성 및 실행

- 저장

- persist(엔티티): insert문이 실행

- 조회

- find(엔티티 타입, 식별자): select문 실행

- 수정

- 특정 메소드 없이 자동으로 변경 감지 후 update: update 문 실행

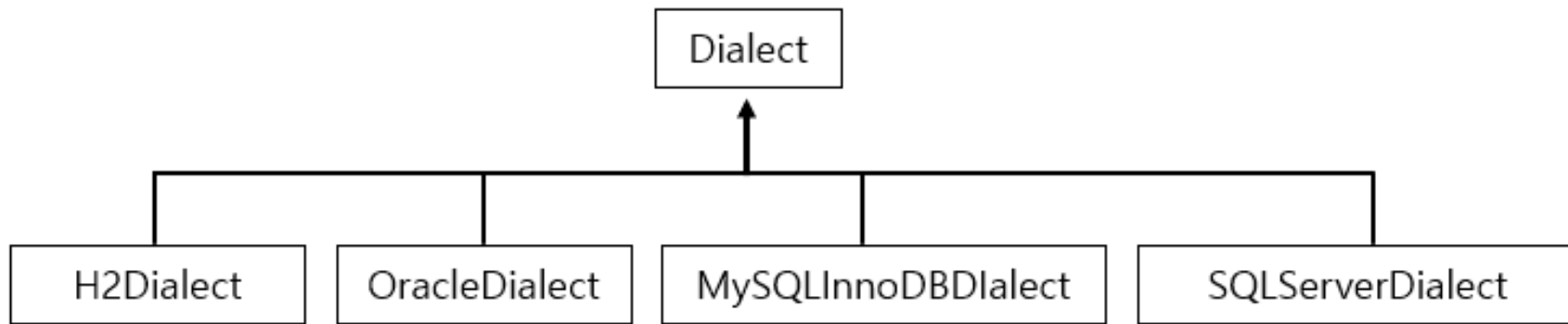
- 삭제

- remove(엔티티)

JPA

• JPA 사용 시 장점

- DB 벤더에 따른 SQL 생성(방언)
 - MySQL → Limit, Oracle → rownum
- JPA는 Dialect라는 추상화된 방언 클래스와 각각의 구현체를 제공



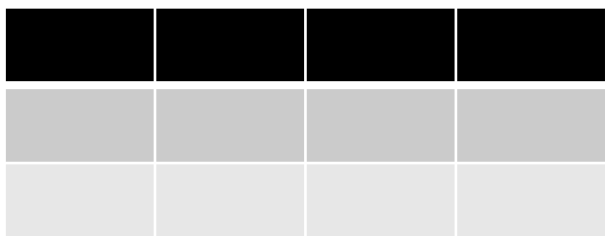
- 데이터베이스가 변경되더라도 SQL 수정 없이 JPA설정만 변경하면 됨

JPA

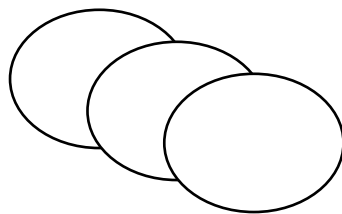
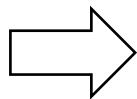
• JPA 사용 시 장점

- Auto DDL기능을 이용하여 테이블 자동 생성

기존



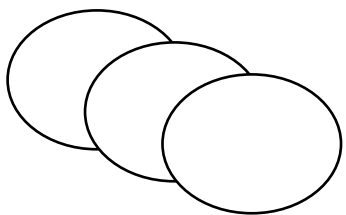
DDL을 통한 테이블 생성



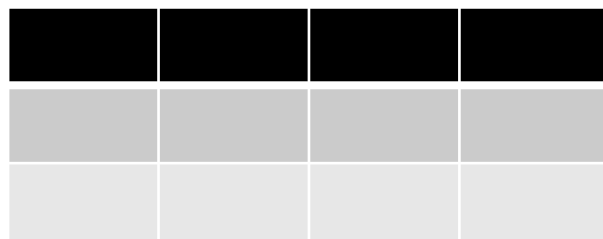
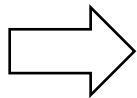
객체 정의

개발단계에서 테이블 수정이 빈번하게
발생한다면?

AUTO DDL



객체 정의



자동생성

JPA

- JPA 사용 시 장점

- SQL을 모아서 한 번에 보내므로 성능 상 이점
- 응용 레벨에서 Repeatable Read 수준의 격리레벨 제공

- 가장 큰 장점은 객체지향 프로그래밍의 설계 원칙을 지키도록 도움

- 이것은 Spring을 사용하는 이유이기도 함

- JPA 단점

- 객체의 응집력을 약화 시킴
- 테이블에 매핑된 엔티티와 별개로 DTO 추가 정의 필요
- 예상치 못한 SQL 발생 가능

Spring Data JPA

- Spring Data JPA

- 스프링 데이터 JPA는 JPA를 편하게 사용하도록 만든 모듈
- JPA(Hibernate)를 사용할 때 발생하는 보일러 플레이트 코드를 제거
- JPA를 사용할 때 발생하는 관습적인 코드 제거
 - 순수 JPA를 사용할 경우, 게시글을 조회하는 findById라는 함수를 만들지만 이는 어쩌면 당연히 있어야 할 예상되는 코드
- CRUD 처리를 위한 공통 인터페이스 제공
- 사용자는 아래와 같이 JpaRepository를 상속하는 인터페이스만 정의

```
public interface MemberRepository extends JpaRepository<Member, Long>
```

추가적으로

- **N+1문제**

- JPA의 성능 튜닝에서 가장 중요한 개념







- **JPQL**

- JPA가 제공해주는 기본 CURD외에 통계성 쿼리 등 복잡한 쿼리를 위해
- JPQL은 트랜잭션 내에서 곧 바로 실행되는 특성이 있으며 이는 N+1문제와 연관이 있음

- **QueryDSL**

- 문자열의 덧셈 연산을 통한 쿼리 작성은 1) 불편하고 2) 오류 발생 가능성이 높음
- 문자열의 덧셈 연산은 특히 동적 쿼리 작성 시 주로 사용
- 동적 쿼리 작성을 간편하고 재사용성이 높은 코드를 만들기 위해 QueryDSL 학습

Persistence frameworks

Logo	Framework Name	Description of the Framework
	Hibernate Framework	Hibernate is an object relational mapping (ORM) framework for the Java programming language.
	Spring Data JPA	Spring Data JPA is used to implement JPA based repositories. It is a part of the Spring ecosystem, and it deals with enhanced support for JPA based data access layers.
	MyBatis	MyBatis is a Java persistence framework and it couples objects with stored procedures or SQL statements using an XML descriptor or annotations.
	EclipseLink	Comprehensive open-source Java persistence solution addressing relational, XML, and database web services.
	Apache OpenJPA	Apache OpenJPA can be used as a stand alone POJO persistence layer or integrated into any Java EE compliant containers.
	Apache Torque	Apache Torque lets you access and manipulate data in a relational database using object relational mapper.