

---

# Introduce Spring Data JPA

---

나는 SQL Hell이 싫어요. 그  
리고 XML도 싫어요.

---

# SQL

---

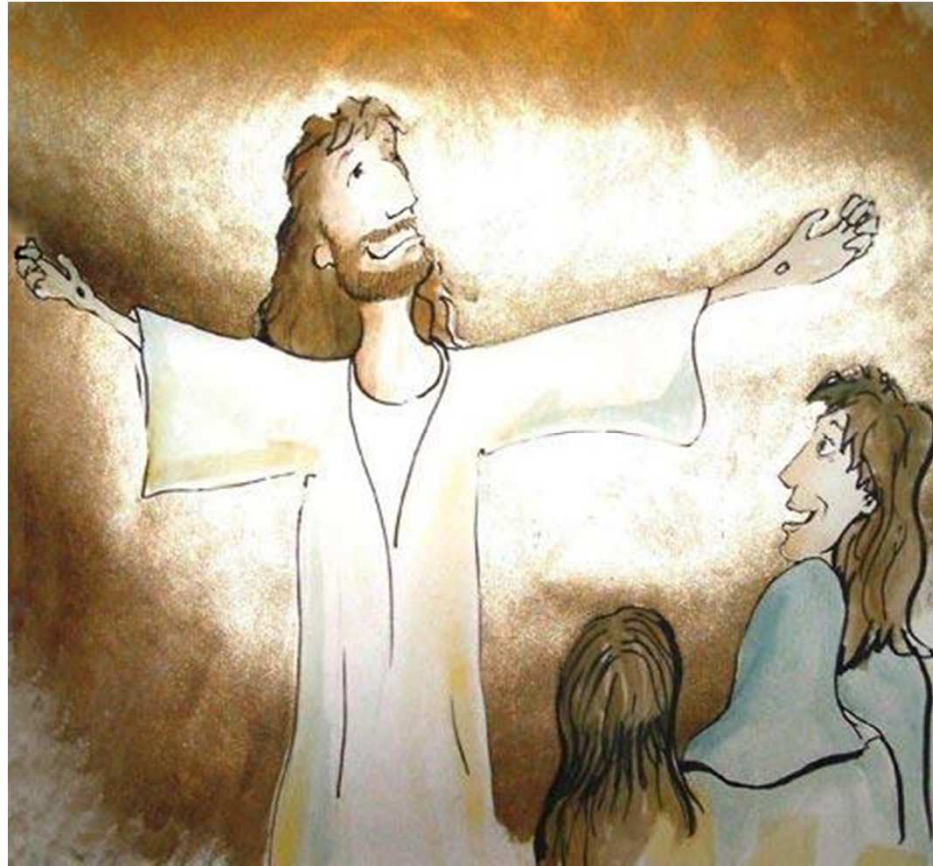
Structure Query Language(구조적 질의 언어)

극악 JDBC 핸들링이 시작되었다.

---

# iBatis가 나를 구원해 주었나?

---



# Spring JdbcTemplate

---

살짝 살려는 줬지만.....

---

# JPA?

---

Java Persistence API

`javax.persistence.*` package!

---

# Spring

---



# Spring Data

---

<http://www.springsource.org/spring-data>

---

# Spring Data JPA

---

- 쉽다.
  - 간편하다.
  - ORM툴에 독립적이다.
-



# Repositories

---

```
public interface Repository<T, ID extends  
Serializable> {  
}
```

---

# Repositories

---

Repository

CrudRepository

PagingAndSortingRepository

JpaRepository

---

# Repository keyword

---

```
public interface EmployeeRepository<Employee, String> {  
    List<Employee> findByName(String name);  
    List<Employee> findByName(String name, Sort sort);  
    List<Employee> findByName(String name, Pageable pageable, Sort sort);  
    List<Employee> findByDepartment(Department department);  
    List<Employee> findByNameOrDepartment(String name, Department ...);  
}
```

---

**Table 2.2. Supported keywords inside method names**

Keyword	Sample	JPQL snippet
And	findByLastnameAndFirstname	... where x.lastname = ?1 and x.firstname = ?2
Or	findByLastnameOrFirstname	... where x.lastname = ?1 or x.firstname = ?2
Between	findByStartDateBetween	... where x.startDate between ?1 and ?2
LessThan	findByAgeLessThan	... where x.age < ?1
GreaterThan	findByAgeGreaterThan	... where x.age > ?1
After	findByStartDateAfter	... where x.startDate > ?1
Before	findByStartDateBefore	... where x.startDate < ?1
IsNull	findByAgeIsNull	... where x.age is null
NotNull, NotNull	findByAge(Is)NotNull	... where x.age not null
Like	findByFirstnameLike	... where x.firstname like ?1
NotLike	findByFirstnameNotLike	... where x.firstname not like ?1
StartingWith	findByFirstnameStartingWith	... where x.firstname like ?1 (parameter bound with prepended %)
ngWith	findByFirstnameEndingWith	... where x.firstname like ?1 (parameter bound with appended %)
Containing	findByFirstnameContaining	... where x.firstname like ?1 (parameter bound wrapped in %)
OrderBy	findByAgeOrderByLastnameDesc	... where x.age = ?1 order by x.lastname desc
Not	findByLastnameNot	... where x.lastname <> ?1
In	findByAgeIn(Collection<Age> ages)	... where x.age in ?1
NotIn	findByAgeNotIn(Collection<Age> age)	... where x.age not in ?1
True	findByActiveTrue()	... where x.active = true
False	findByActiveFalse()	... where x.active = false

# Entity

---

- @Entity로 선언
  - Annotation으로 Entity간 relation을 설정한다.
-

# 코드로 말해요

---

springmvcjpa 프로젝트

---

# @Query

---

JPQL

```
@Query("SELECT e FROM Employee a  
WHERE a.name = ?1")
```

---

# 정리

---





# 결론

---

