## Agenda

1) Implement Self productService
2) Query Methods
   - → Declared Query Methods
   - → Hibernate Query Methods
   - → SQL
3) Fetch Types
   - → Eager
   - → Lazy

## Next Class

① Representing Cardinalities
   - → MappedBy
   - → Cascading

② Schema Versioning

# JPA Repository

[ ———— findbyId( ); ] Interface

⇓

[ Select * from <<Table>> where id = <<id>> ; ]

and execute on DB

# Declared Queries

=> No need to write SQL queries on our own

=> Just write the method name & ORM will convert that method name into corresponding query

findby Id ( )

{ fetch by Id
Query by Id

Product → Category

{ Select * from product
where id = "id"
join * from Category where id = cat_id

{ id
name
title
Category-id

Product {
id
name
title
} category_id

Client ←

(Category) (category_id)

Entity → dependends on another (Entity)

JOIN

EAGER

**Eager fetching** : fetch the details of inner object along with Outer Object

Category
└→ List<Products>

{ Select * from category
  join Select * from products
    where CategoryId = "iq"

Category → [10k Products] 1% scenarios

**Lazy fetch** : Don't fetch the details of inner object along with outer object

Only fetch when required

{ [category.getProducts()]

[ Outer Object [One]
  └→ Inner Object ] → Eager

Outer object
  └→ List<Inner Object> ] → Lazy