



JPA ORM 6단원

bangjang



Entity 매핑 3요소

1. 다중성
2. 단방향 / 양방향
3. 연관관계의 주인

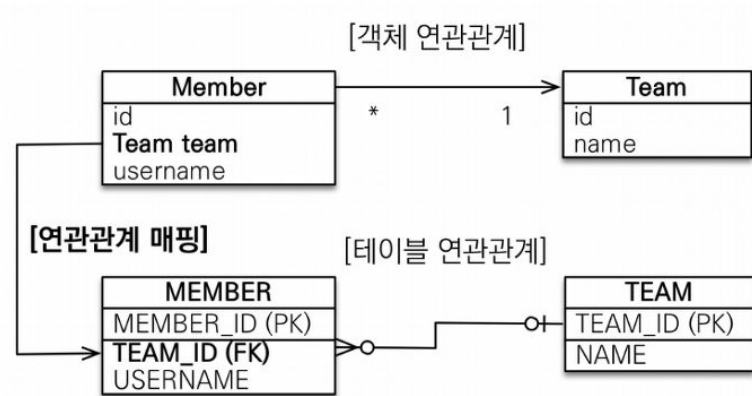
@ManyToOne (다대일) - 단방향

N : 1

회원은 Team 엔티티 참조 가능

Team은 회원 참조 불가능

@JoinColumn을 통해 필드를 외래키와 매핑

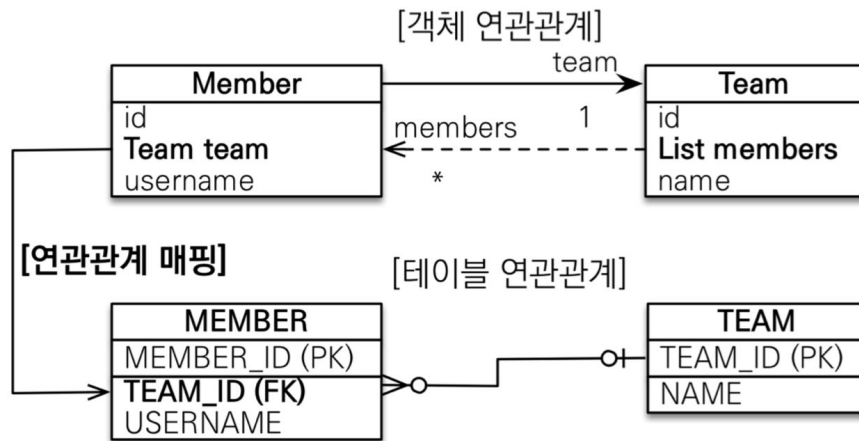


@ManyToOne (다대일) - 양방향

N : 1, 1 : N

N에 외래 키가 있음 (연관 관계의 주인)

항상 서로를 참조해야 함



@OneToMany (일대다) - 단방향

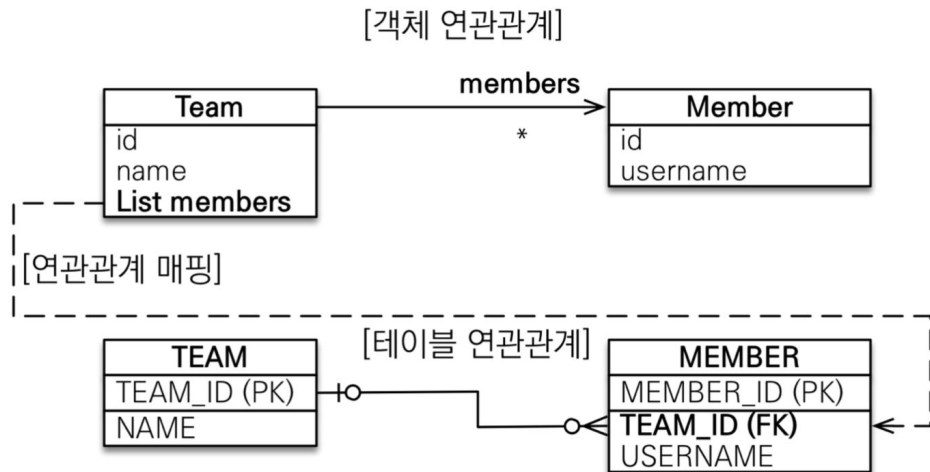
1:N

다대일의 반대 방향

엔티티를 여러 개 참조 가능, Java Collection

@JoinColumn 명시 필요

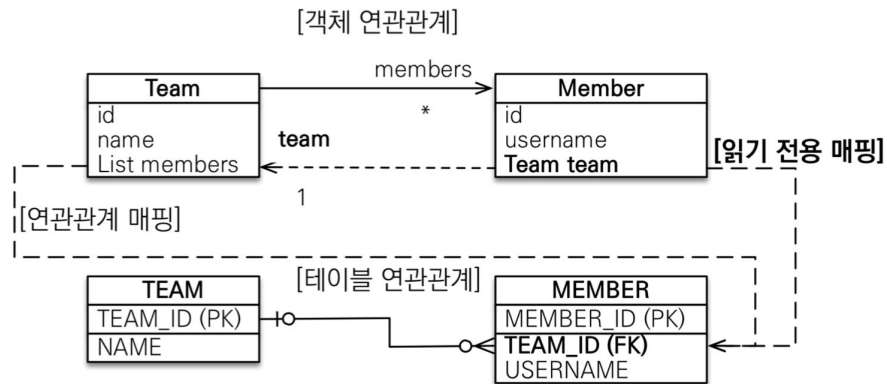
타 테이블 외래 키 관리 문제, 권장되지 않음



@OneToMany (일대다) - 양방향

존재하지 않는 매핑

다대일 양방향과 같음 - 다대일 양방향 권장





@OneToOne (일대일)

양쪽에 하나의 관계만 가짐 (회원과 사물함)

주 테이블, 대상 테이블 어느 곳이든 외래 키 가질 수 있음

주 테이블에 외래 키 : 주 테이블에 키를 두고 대상 테이블 참조, 객체지향 선호

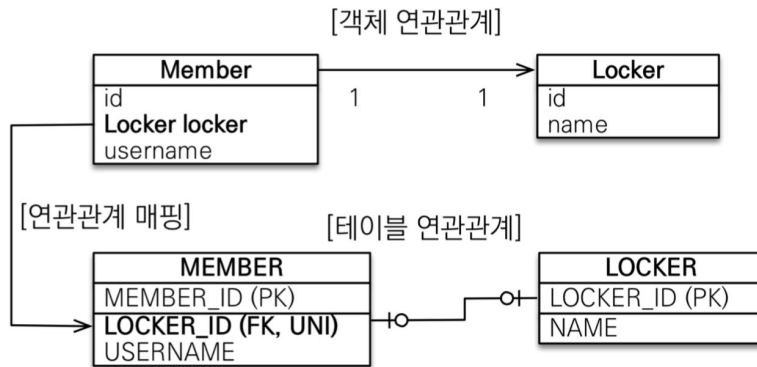
대상 테이블에 외래 키 : 전통적 방법, 일대다 변경 시 테이블 구조 유지 가능

@OneToOne (일대일) - 주 테이블에 외래 키 (단방향)

@OneToOne 매핑

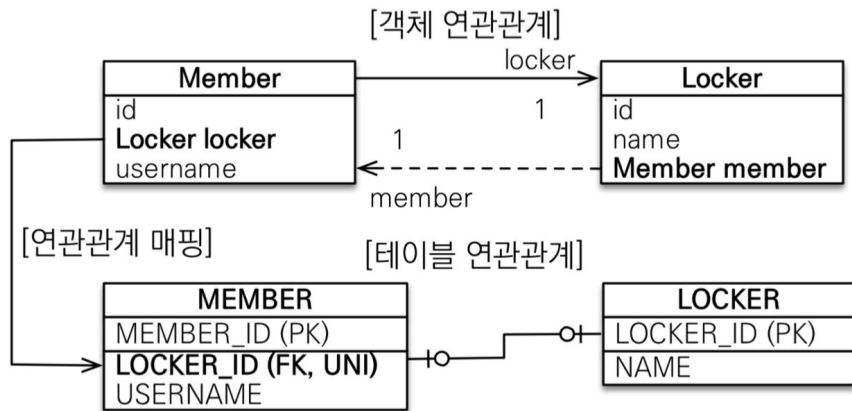
Member에서 참조

다대일 단방향과 비슷



@OneToOne (일대일) - 주 테이블에 외래 키 (양방향)

연관관계 주인 - 외래 키 테이블

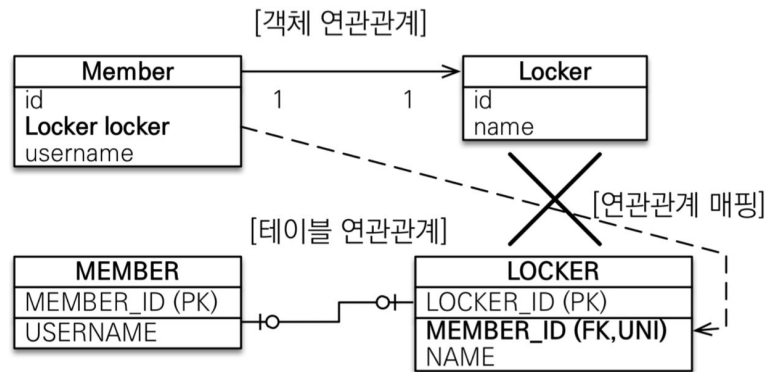


@OneToOne (일대일) - 대상 테이블에 외래 키 (단방향)

대상 테이블 외래 키 단방향은 JPA 미지원

매핑 방법이 없음

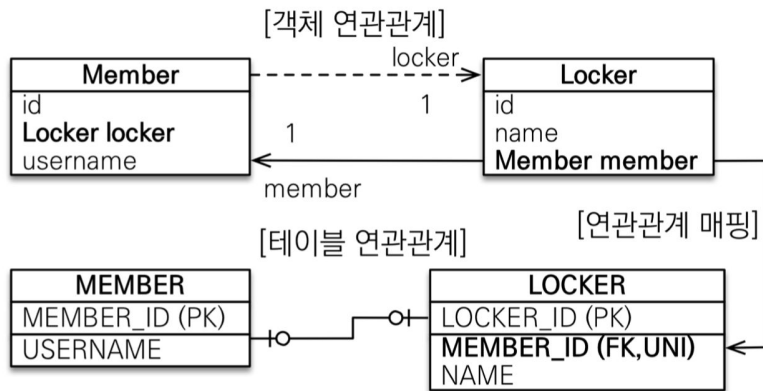
Locker -> Member로 수정 / 양방향으로 변경



@OneToOne (일대일) - 대상 테이블에 외래 키 (양방향)

일대일 매핑에서 대상에 외래 키 - 양방향 매핑

대상 테이블인 Locker가 주인이 되어 외래 키 관리



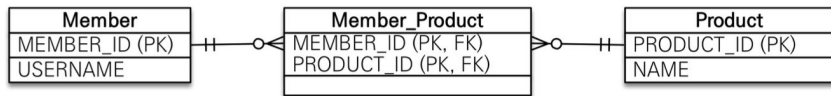
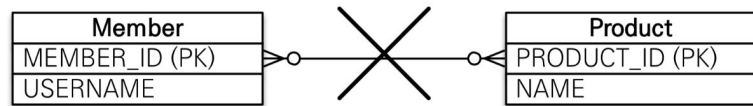
@ManyToOne (다대다)

RDB는 정규화된 테이블 2개로 다대다 표현 불가능

다대다를 일대다, 다대일로 풀어내는 연결 테이블 사용

객체는 @ManyToOne를 사용

연결 테이블 없이 간단한 다대다 생성 가능



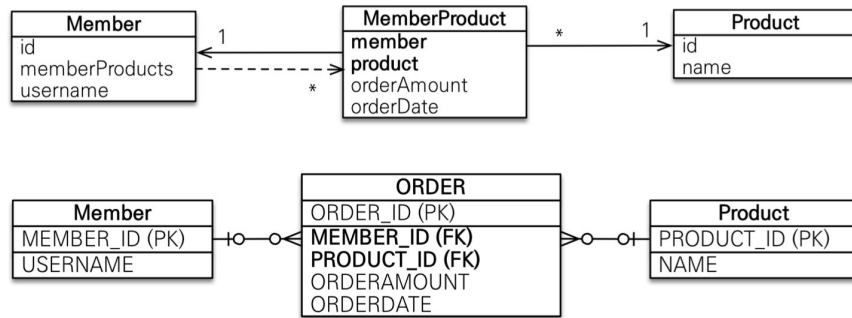
@ManyToMany (다대다) - 연결 엔티티 사용

@ManyToMany의 실무 한계 - 추가된 컬럼 매핑 X

테이블 연결 매핑 엔티티 생성 후 매핑 - 일대다, 다대일

부모 테이블 기본 키를 받아 기본 키로 사용 - 식별 관계

복합 키 - ORM 매핑 복잡해짐





@ManyToMany (다대다) - 연관관계 정리

일대다, 다대일로 풀기 위해 식별자 구성 선택 필요

- 식별 관계 : 받아온 식별자를 기본 키 + 외래 키로 사용
- 비식별 관계 : 식별자를 외래 키로만 사용하고 새로운 식별자 추가

객체지향적으로 비식별 관계가 편리