

## 9-3. 영속성 전이와 고아 객체

### 영속성 전이

부모 엔티티를 영속성 컨텍스트에 저장할 때 자식 엔티티도 같이 저장하고 싶을때!!

하나의 부모만 자식들을 관리할 때. 다른 데서도 자식들과 연관관계가 있다면? 안됨!

ex) 게시판에 첨부파일의 경로, 첨부파일의 데이터가 들어가있을 때 가능.

→ 게시판이 부모. 첨부파일의 경로, 데이터가 자식.

=> 부모와 자식의 라이프사이클이 동일할 때와 단일 소유자일 때 사용해도됨

### 고아 객체

부모 엔티티와 연관관계가 끊어진 자식 엔티티를 자동으로 삭제

`orphanRemoval = true` 로 사용 가능

=> 참조가 제거된 자식 엔티티를 다른 곳에서 참조하지 않는 고아 객체로 취급하여 삭제

참조하는 곳이 하나일 때, 단일 소유자일 때 사용!

또, 고아 객체 설정을 하면 부모 삭제 시 자식 엔티티도 모두 삭제

=> delete 쿼리가 계속 나간다

단일 소유자일때만 사용하자! 조심히 사용

### 영속성 전이 + 고아 객체

```
em.persist(parent);  
em.remove(findParent);
```

위처럼 부모만 지워도 자식들이 지워짐.

parent의 생명주기는 JPA가, child들의 생명주기를 parent가 관리함.

따라서 child에는 뒤에있는 리포지토리가 필요가없다..?

DDD의 Aggregate Root를 사용할 때 유용함.

Aggregate Root : DDD에서 리포지토리를 Aggregate Root가 관리하는 밑에 있는 다른 것들은 리포지토리를 만들지 않는다!

#JPA