Sec04

JPA 에서 @ 사용하기

작성자: 강태경



# Exam03

JPA 페이징 처리 예제

* 페이징 처리 함수  
   목적 : 전체 데이터를 화면에 출력하면 성능과 가독성이 떨어지므로 , 몇건씩 끊어서 보여주는것이  
   목적  
   속성 : page = 현재페이지, size = 1 페이지 당 보여줄 데이터 수 ( url 매개변수 전달 )  
   페이징 객체 : 1) 매개변수 페이징 객체 : Pageable  
   2) 리턴될 페이징 객체 : Page<객체자료형>  
   클라이언트로 전송할 데이터 : Map 자료구조 이용  
   1) 데이터 : 부서, 사원 등  
   2) currentPage : 현재 페이지 수  
   3) totalItems : 전체 데이터 수  
   4) totalPages : 전체 페이지 수

# Exam04

JPA에서 조인(관계 설정) @ManyToOne(\*), @OneToMany, @OneToOne  
 ( 추천 안함: @ManyToMany ) 이 있음,  
단방향 조인 ( 추천 ) -> 부서클래스 @OneToMany 달지 않음, 사원클래스 @ManyToOne  
양방향 조인 ( 필요시 ) -> 부서클래스 @OneToMany 달기 , 사원클래스 @ManyToOne

* @ManyToOne : N : 1 관계에서 N인 클래스에 사용   
  JPA : 자동 sql 문 생성 -> FK(외래키) 만들어 줌
* @ManyToOne기본 fetch 옵션은 성능 저하가 발생 : 몇가지 문제가 발생함  
  1) N + 1 문제 : 1st 쿼리 하나당 2nd 쿼리가 또 실행되어 성능이 대폭 저하되는 문제  
  > 해결책 : @ManyToOne(fetch = FetchType.LAZY) 사용(지연로딩),  
   조인 속성 출력 시 @Query(sql문) 에서 join fetch 사용

2) 잭슨(Jackson) : 객체 -> JSON 자동변환 기능에서 문제 발생  
스프링에서 (클래스) 객체를 json 데이터로 변환해서 byte로 클라이언트 전송( REST API, @RestController )  
> 해결책 : @JsonIgnore, @JsonBackReference

적용 예제 )  
@ManyToOne(fetch = FetchType.LAZY)  
@JoinColumn(name = "dno")  
@JsonBackReference  
private Department department;

* 양방향 관계 설정 시 사용 :   
  @OneToMany(mappedBy = "상대클래스FK속성명")//   
  단, 단방향 조인(FK) 설정 시 설정하지 않음

주로 단방향으로 걸고, 꼭 필요할 경우 양방향으로 관계 설정을 함

속성 타입은 주로 List, Set 으로 만듬   
( Many 쪽과 연결하므로 배열로(List) 설정 )

실제 테이블에는 아래 속성이 컬럼으로 생성되지 않음   
( 양방향 연관관계 설정만 하는 것, 자바 클래스에서만 속성 코딩 )

사용법 : @OneToMany(mappedBy = "상대 클래스의 @ManyToOne 이 붙은 속성명)

* 양방향 시 문제점 : Lombok @ToString 을 걸면 무한루프 발생 제거 필요
* 해결책:   
  1) DTO 클래스를 만들어서 해결  
  2) ToString(exclude=”employee”)  
  2) @JsonBackReference 달기(@ManyToOne 이 있는 클래스)   
   @JsonManagedReference 달기(@OneToMany 가 있는 클래스)  
  적용예제)  
  @OneToMany(mappedBy = "department")  
  @JsonManagedReference  
  private List<Employee> employee = new ArrayList<>();