

JPA - 4 장. 엔티티매핑

매핑 정보는 XML 과 어노테이션중에 선택 할 수 있는데 어노테이션을 사용하는 쪽이 좀 더 쉬움

어노테이션은 크게 4 가지로 분류

- 객체와 테이블 매핑
- 기본 키 매핑
- 필드와 컬럼 매핑
- 연관관계 매핑

객체와 테이블 매핑

- @Entity
 - 테이블과 매핑할 클래스에 붙이는 어노테이션
- @Table
 - 엔티티와 매핑할 테이블을 이름으로 사용 (@Table(name = 'tableName'))

기본키 매핑

- @Id
- @GeneratedValue(strategy = GenerationType.IDENTITY)
 - 시퀀스 번호 자동생성

필드와 컬럼 매핑

- @Column
 - 객체 필드를 테이블 컬럼에 매핑
 - 생략가능, 속성기본값이 적용됨
- @Enumerated
 - 자바의 enum 타입을 매핑할때 사용
- @Temporal
 - 날짜 타입을 매핑할때 사용

- @Lob
 - 데이터베이스 BLOB, CLOB 타입과 매핑
 - @Transient
 - 해당 필드는 데이터베이스와 매핑하지않음
 - @Access
 - JPA 가 엔티티 데이터에 접근하는 방식을 지정
-

데이터베이스 스키마 자동생성

JPA 는 데이터베이스 스키마를 자동으로 생성하는 기능을 지원

- Application.properties 에 아래 내용 추가

```
#ddl 생성
spring.jpa.hibernate.ddl-auto=create
```

```
#console 에 sql 보여줌
spring.jpa.show-sql=true
spring.jpa.properties.hibernate.format_sql=true
```

- ddl-auto 의 속성
 - Create : 기존 테이블을 삭제하고 새로 생성
 - Create-drop : create 속성에 추가로 애플리케이션을 종료할 때 생성한 DDL 을 제거
 - Update : 데이터베이스 테이블과 엔티티 매핑정보를 비교해서 변경사항만 수정
 - Validate : 데이터 베이스 테이블과 엔티티 매핑정보를 비교해서 차이가있다면 경고 남김
 - None : 자동 생성 하지 않음