

# [블로그] DB 설계는 어떻게 해야 할까?

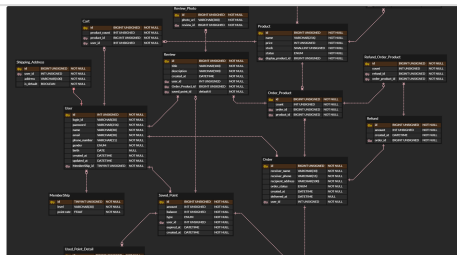
☀ 상태	완료
☰ 다중 선택	데이터베이스
🕒 생성 일시	@2024년 4월 23일 오후 8:33

## Reference

DB 설계는 어떻게 해야 할까?

도대체 어떻게 설계해야 할까? 🤔

<https://velog.io/@sontulip/how-to-db-design>



## 좋은 데이터베이스 설계란?

1. 무결성 - 데이터베이스 내에 모든 값은 언제나 정확한 값을 유지해야 한다.
2. 유연성 - 데이터베이스 구조는 요구사항 변화에 대해 수정이 쉬워야 한다.
3. 확장성 - 데이터베이스 구조는 기능 확장에 대해서 수정이 쉬워야 한다.

이런 테이블 설계 목표들이 '돈' 💰 을 벌어들여주더라!

## 테이블 설계

### 1. 예비 테이블 설계

- 필요한 필드들이 뭐가 있을지 예비 필드 목록을 뽑고 적절히 묶어서 예비 테이블을 만든다.

### 2. 이상적인 테이블 구조에 따라 테이블 구성하기



- 테이블은 **단일 주제** 를 나타내야 한다.

- 단일 주제는 개체(유저, 상품, 후기 등) 또는 사건(주문 기록, 결제 기록 등)이다.
- 기본 키를 가진다.
  - ▼ 인조키 vs. 자연키
    - 자연키는 id(로그인할 때 사용하는 거!), email, 주민번호 등이 예시
    - 인조키는 id(int)가 예시
    - 자연키는 **성능** vs. 인조키는 **유연성**
- 다중값 또는 다중 부분 필드를 포함하지 않는다.
  - 다중값 필드 → 관계로 해결
  - 다중 부분 필드 (ex. 주소) → 기획에 따라 다르다.
- 계산된 필드들을 포함하지 않는다.
- 외래키 외에 불필요하게 중복되는 필드들을 포함하지 않는다.
- 절대적으로 최소한의 중복데이터만 포함한다.
 

ex. 등급테이블을 따로 둘 수 있다.

### 3. 필드 명세 설정

각 필드마다 이름, 데이터 타입, 길이, 범위, 유일성, null 허용, 기본값을 어떻게 할지를 정해야 한다.

- char vs. varchar
  - CHAR는 저장공간이 낭비되고
  - VARCHAR는 수정에 대한 성능이 떨어진다
  - 삽입보다는 조회 성능이 중요하다 → varchar
- datetime vs. timestamp
  -  MySQL 날짜 및 시간 데이터 타입 비교
- enum vs. 참조 테이블
  - enum : 성능 / 간편함
  - 참조테이블: 유연성
  - 자주 변경이 되는가? 다른 데이터들이 추가될 여지가 있는가?
  -  MySQL Enum

## 4. 관계 설정

- 일대일, 일대다, 다대다
- 다대다 관계는 매핑테이블을 사용해서 일대다 관계로 해소
- 식별 vs. 비식별

## + 추가 고민

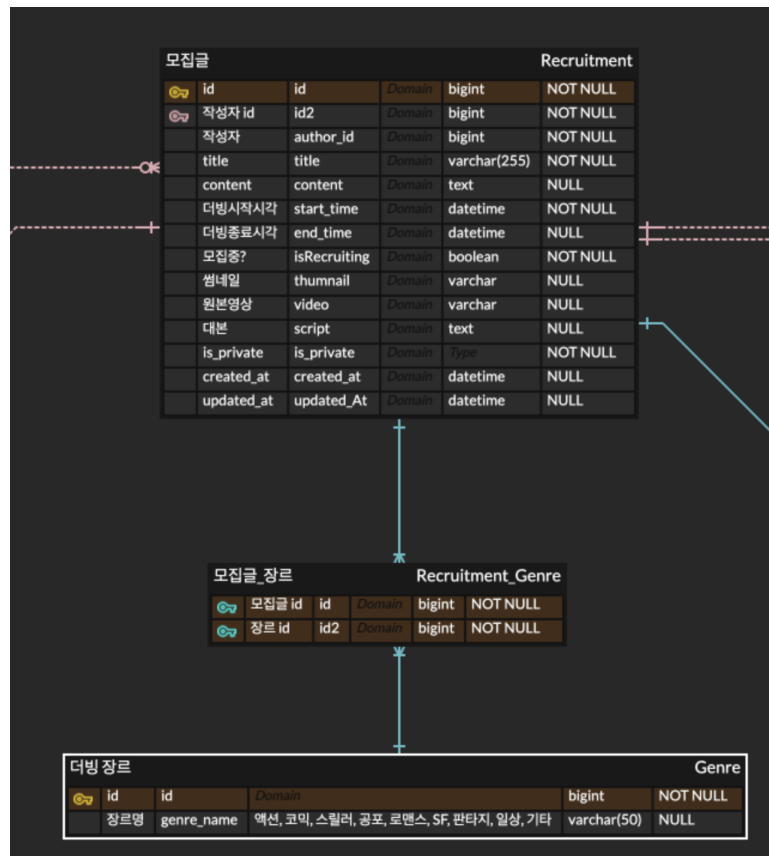
- ~외 4종 상품은 어떻게 처리할 것인가?
- 카테고리 3, 4, 5 덤스로 이어진다면?
- 1년 이상된 적립금 만료 기능은?
- 부분 환불 ?
- 상품 할인 요구사항에 더해서 정액 할인 요구 사항이 생긴다면?

## + 정규화 / 반정규화

---

## ▼ 250123 ERD 피드백

1. ERD를 왜 작성해야 하는가?
  - a. 관계자들과의 소통/이해
2. 순환참조
  - 데이터 무결성에 문제
  - fk를 많이 들고가지 않는다.
3. 그러면 식별관계는 많이 하시나요?
  - 오늘 erd에서는 명확하다. 유니크하다. 변동이 없다.
  - 인조키를 가져갈 때 데이터 정합성에 문제가 발생할 일이 생길 수 있다.
  - 그럼에도 jpa를 사용하면 복합키를 사용하기가 까다롭다. → 인조키를 만들고 복합키를 unique 설정을 해줄 수 있다.



## ERD Space

ERD Tool, Database Modeling Tool, create ER diagrams and export ddl, pdf

<https://www.erdspace.com/>