



# 빈 미션 페이지

## 핵심 키워드

### ▼ 외래키

MySQL 기준 Foreign Key,

한 테이블에서 다른 테이블의 기본 키를 참조, 사용하는 키, 말 그대로 외래종 같이 다른 테이블의 키예요.

외래키를 사용한다는 것은 다른 테이블과 연관되어 있다라고 볼 수 있어요. A 테이블의 한 레코드 값이 변경되었는데 B 테이블에서는 변경되지 않았다면? 데이터의 무결성에 매우 위반되므로, 주로 무결성을 지키는 데 사용해요

### ▼ 기본키

MySQL 기준 Primary Key, PK.

공백이면서 최소이자 유일한 성격을 띄며, 한 레코드를 식별할 때 사용되는 필드나 집합을 뜻해요.

각 테이블의 레코드는 무조건 이 기본 키를 가지고 있어야 해요.

### ▼ 복합 키

MySQL 기준 Composite Key,

두 개 이상의 컬럼으로 이루어진 기본 키를 뜻해요.

기본키를 무조건 가지면서, 2개 이상을 가질 수도 있기에, 설정된 컬럼이 유일성과 최소성을 띄어야 해요.

이 복합키는 외래키 처럼 데이터 무결성과 정규화에 도움을 줘요

### ▼ ER 다이어그램

한글 까지 줄여서 ERD, Entity, Relations Diagram, 말 그대로 엔티티 관계 다이어그램!

DB 설계를 위해 팀원들과 소통하며 그리는 다이어그램 그림이라고 해요.

툴로는 ER Cloud, AQueryTool, Draw.io 등이 있어요.

## ▼ 연관관계

테이블, 엔티티 간에 존재하는 관계를 뜻해요.

종류로는 1 : 1, 1 : N, N : 1, N : M → 일대일, 일대다, 다대일, 다대다 관계가 있어요.

## ▼ 정규화, Normalization

ER 다이어그램 내에 중복 요소를 제거하는 세 단계를 걸치는 과정이에요

총 세 단계가 있지만, 공통적인 목적은 **중복된 속성을 최소화** 하고, **종속 관계에 있는 속성을 제거**하는 과정이에요, 이 과정으로 중복 데이터를 줄이고 관계형 데이터베이스 구조를 보다 효율적으로 설계할 수 있어요!

### 1. 제 1 정규화

- 같은 성격, 같은 내용의 컬럼이 연속적으로 나타날 때가 발생할 수 있어요
- 컬럼을 제거하고, 기본 테이블에 PK를 추가한 새로운 테이블을 만들어 연관관계를 지어요. (1 : N)

### 2. 제 2 정규화

- 복합키가 존재하는 경우가 발생할 수 있어요
- 이를 분리하여 새로운 테이블을 만들고 분리하여, 외래키로 연결하여 연관관계를 지어요.

### 3. 제 3 정규화

- 기본키를 만들었지만, 정작 다른 컬럼에 의존하는 경우가 발생할 수 있어요
- 이 또한 컬럼을 테이블화 시켜 외래키로 연결하여 연관관계를 지어요.

## ▼ 반 정규화

정규화와 다르게 일부러, 의도적으로 정규화 단계, 원칙을 어기는 행위를 말해요. 정규화 또한 장점 투성이 처럼 보이지만, 단점이 존재해요.

컬럼을 독립시켜 테이블화 하고 연관관계를 지는 과정이 해결 방안인데, 만약 엔티티가 엄청 커지면....? 연관관계 Join 하는 과정이 엄청 많아 DB 응답 시간이 느려질 수 있어요

그래서 약간의 ER 모델의 연관관계, 유연성의 구조는 낮추고 중복 데이터를 어느 정도 허용하는 리스크를 택한거죠. 대신 DB 조회 속도와 성능을 높힐 수 있어요.

## 기본 세팅

### 1. 테이블, 칼럼 이름 모두 소문자, 언더 바로

2. 기본 키: id bigint

3. created\_at, updated\_at datetime(6) 이런 형식으로 생성, 업데이트 날짜 → Hard Delete 방지, Soft Delete, Batch!

---

## 요구사항

### 사용자

로그인, 회원가입

- 약관 동의에서 필수 약관과 선택 약관이 나뉜다.
- 회원가입은 기본적으로 모두 SNS 로그인을 통해 이뤄진다.
- 이름, 성별, 생년월일, 주소는 필수 입력이다.
- 선호 음식 종류를 최대 2개 선택할 수 있다.

마이페이지

- 닉네임, 이메일, 전화번호, 내 포인트, 작성 리뷰, 문의하기 등을 할 수 있다.

### 리뷰, 문의

- 한 사용자가 여러 리뷰, 여러 문의를 남길 수 있다.
- 리뷰에 대한 중복 사항이 많아 유저와 음식점에 대한 중간 테이블화 하였다.

### 미션

- 한 사람은 여러 미션은 할 수 있다. 한 미션에 여러 사람이 할 수 있다.
- 미션을 만들 때 중복되는 요소가 많아 중간 테이블화 시키고, 음식점 테이블을 연결한다.

### 사진

- 실제 파일 통째로 저장하는 것이 아님! 사진 경로를 저장하는 것!

## ERD

<https://www.erdcloud.com/d/NLcqCfTbFY8ygum65>

- many 관계선은 erdCloud 기준 **zero or many**
- 많은 쪽에, N 쪽에 화살표

## 관계 형성 및 정규화

### ▼ 기본 테이블 구현 과정

1. 한 사용자는 여러 미션을 할 수 있고, 한 미션도 여러 사용자가 할 수 있다 → N : M 관계
  - a. user → user\_mission ← mission
2. 사용자는 여러 음식 태그를 고를 수 있다 (최대 2개). 한 태그에 여러 사용자가 올 수 있다 → N : M 관계
  - a. user → user\_food\_tag ← food\_Tag
3. 음식점은 한 음식 태그만 고를 수 있다. 반대로 태그에는 여러 음식점이 올 수 있다. → N : 1 관계.
  - a. food\_tag → restaurant
4. 사용자는 여러 리뷰, 문의를 남길 수 있다. 하지만 한 리뷰, 문의에 여러 사용자는 이루어질 수 없다. → 1 : N 관계
  - a. user → review
  - b. user → inquiry
  - c. 한 사용자에게 여러 리뷰, 한 레스토랑에 여러 리뷰가 올 수 있다.  
but 한 리뷰에 여러 사용자 여러 레스토랑 은 이루어질 수 없다.
5. 음식점에는 여러 리뷰를 남길 수 있다. 하지만 한 리뷰가 여러 음식점은 이루어질 수 없다. → 1 : N 관계
  - a. restaurant → review
  - b. user → review ← restaurant
6. 한 음식점에 여러 미션이 있다. 한 미션에 여러 음식은 이루어질 수 없다. → 1 : N 관계
7. 사용자는 여러 포인트 이력을 남길 수 있다. 하지만 한 포인트 이력에 여러 사용자는 이루어질 수 없다. → 1 : N 관계

8. 한 사용자에게 대해 여러 알람 설정이 있다. 한 알람에 대한 여러 사용자?  $X \rightarrow 1:N$  관계

a. user  $\rightarrow$  alarm

- 
- 리뷰는 사용자가 남기면서, 레스토랑에서도 접근 가능하기에 불필요한 중복이 생긴다.  
제 1 정규화를 통해 리뷰를 테이블화 하고, 중복 제거한다.
  - 사용자 - 미션 간 중간 테이블인 user\_mission 테이블에서는 연결 관계만 형성해야 한다.  
 $\rightarrow$  제 2 정규화를 통해 미션 관련 진행 여부인 complete 컬럼을 mission 테이블로 옮긴다.

## 테이블

사용, user

- id
- SNS 로그인 방식?
- 이메일
- 이름
- 닉네임
- 전화번호
- 전화번호 인증 여부
- 성별 (남 여 선택X)
- 생년월일  
(yyyyMMdd)
- 주소
- 포인트
- 생성 일자,  
datetime(6)
- 업데이트 일자,  
datetime(6)

음식점

- id
- 음식점 이름
- 음식 종류
- 주소
- 영업 여부

미션, mission

- id
- 미션 내용
- 미션 유효 기간
- 선행 결제 금액
- 미션 성공 포인트
- 미션 생성 일자
- 미션 업데이트 일자

- inactive\_date,  
datetime

#### 문의, inquiry

- id
- 문의 제목
- 문의 내용
- 문의 사진
- 문의 시간
- 답변 여부

#### ~~포인트 아력~~

- ~~id~~
  - ~~미션 완료한 사용자~~
  - ~~미션 완료 시간~~
  - ~~자급 포인트~~
- 중간 테이블로 대체

#### 리뷰

- id
- 별점
- 리뷰 내용
- 사진
- 음식점 답변 내용

#### 알림 및 서비스? (1:1 관계)

- id
- 위치 정보 제공 여부
- 마케팅 수신 동의 여부
- 새 이벤트 수신 여부
- 리뷰 답변 알림 여부
- 문의 내역 답변 알림 여부

#### 태그

- id
- 음식 종류 (varchar)

#### 알림

- id
- 알림 dtype
- 제목
- 내용
- 생성 일자
- 업데이트 일자

## 결과 다이어그램

<https://www.erdccloud.com/d/NLcqCfTbFY8ygum65>

