

## 프로젝트 3: 관계형 스키마를 이용한 논리적 (logical) DB 설계

2018007965 김산

지난 과제인 프로젝트 2에서 누락된 부분이 너무 많아서 이번 프로젝트3 과제를 하면서 좀 추가를 했습니다.

<추가 사항> (Entity – Attribute)

ADMINISTRATOR – ID (PRIMARY KEY)

ALBUM – ID (PRIMARY KEY)

ARTIST – ID (PRIMARY KEY)

PLAYLIST – ID (PRIMARY KEY)

PLAYLIST – NAME

<수정 사항>

MUSIC의 ID를 PRIMARY KEY로 지정하였습니다.

USER의 ID를 PRIMARY KEY로 지정하였습니다.

GENRE의 NAME을 PRIMARY KEY로 지정하였습니다.

중복되는 이름을 가진 RELATIONSHIP들의 이름에 관계된 두 ENTITY의 이름들을 추가해서 중복이 안 되게 하였습니다.

<Logical DB 설명>

- ADMINISTRATOR
  - ID라는 PRIMARY KEY를 가지고 있다.
- USER
  - NAME, PASSWORD, NICKNAME, GENDER를 가지고 있다.
  - ID라는 PRIMARY KEY를 가지고 있다.

- MNG\_ID는 ADMINISTRATOR의 ID를 참조하는 FOREIGN KEY이며 MANAGE RELATIONSHIP의 1대N 관계를 나타낸다.
- SIGN\_ID는 ADMINISTRATOR의 ID를 참조하는 FOREIGN KEY이며 SIGN UP RELATIONSHIP의 1대N 관계를 나타낸다.
- PLAYLIST
  - NAME을 가지고 있다.
  - ID라는 PRIMARY KEY를 가지고 있다.
  - UID는 USER의 ID를 참조하는 FOREIGN KEY이며 HAS (USER – PLAYLIST) RELATIONSHIP의 1대N 관계를 나타낸다.
- MUSIC
  - NAME, INFO, LYRICS, COMPOSE를 가지고 있다.
  - ID라는 PRIMARY KEY를 가지고 있다.
  - ALBUMID는 ALBUM의 ID를 참조하는 FOREIGN KEY이며 BELONG TO (ALBUM – MUSIC) RELATIONSHIP의 1대N 관계를 나타낸다.
  - ENROLL\_ID는 ADMINISTRATOR의 ID를 참조하는 FOREIGN KEY이며 ENROLL MUSIC RELATIONSHIP의 1대N 관계를 나타낸다.
  - DELETE\_ID는 ADMINISTRATOR의 ID를 참조하는 FOREIGN KEY이며 DELETE MUSIC RELATIONSHIP의 1대N 관계를 나타낸다.
- ARTIST
  - NAME, DEBUT, COMPANY를 가지고 있다.
  - ID라는 PRIMARY KEY를 가지고 있다.
- ALBUM
  - NAME, COMPANY, RELEASE를 가지고 있다.
  - ID라는 PRIMARY KEY를 가지고 있다.
- GENRE
  - NAME이라는 PRIMARY KEY를 가지고 있다.
- SHARE

- USER와 PLAYLIST의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
- UID는 USER의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- PID는 PLAYLIST의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- PREFERRED MUSIC
  - USER와 MUSIC의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - UID는 USER의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - MID는 MUSIC의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- PREFERRED ARTIST
  - USER와 ARTIST의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - UID는 USER의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - ARTISTID는 ARTIST의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- PREFERRED GENRE
  - USER와 GENRE의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - UID는 USER의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - GNAME은 GENRE의 NAME을 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- BELONG TO (PLAYLIST – MUSIC)
  - PLAYLIST와 MUSIC의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - MID는 MUSIC의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - PID는 PLAYLIST의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- HEARD
  - USER와 MUSIC의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - UID는 USER의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - MID는 MUSIC의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- HAS (ARTIST – GENRE)
  - ARTIST와 GENRE의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - ARTISTID는 ARTIST의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.

- GNAME은 GENRE의 NAME을 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- HAS (ALBUM – GENRE)
  - ALBUM과 GENRE의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - ALBUMID는 ALBUM의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - GNAME은 GENRE의 NAME을 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
- HAS (MUSIC – GENRE)
  - MUSIC과 GENRE의 M대N 관계를 나타내는 RELATIONSHIP을 표현한 테이블이다.
  - MID는 MUSIC의 ID를 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.
  - GNAME은 GENRE의 NAME을 참조하는 FOREIGN KEY이자 PRIMARY KEY이다.

