

DB_Project_3



Database Systems

관계형 모델을 이용한 논리적 DB 설계

2016025469

서건식

관리자

계정번호	도영	식별 ID	원등록번호	이름	성별	부서 식별번호	등록-음악ID	삭제음악ID
------	----	-------	-------	----	----	---------	---------	--------

핸드폰 번호
관리자 식별 ID
핸드폰번호

부서
식별 번호
부서이름
부서-관리자ID

부서위치
부서 식별 번호
부서 위치

관리영역
부서 식별 번호
관리영역

음악

제목	아티스트	장르	가사	플레이션	등록일자	발매일자	앨범제목	유저비디오	순위	음악ID
----	------	----	----	------	------	------	------	-------	----	------

플레이리스트
소유한 사용자
제목

플레이리스트-등록
플레이리스트 등록-음악ID
플레이리스트-소유한 사용자

플레이리스트-삭제
플레이리스트 삭제-음악ID
플레이리스트-소유한 사용자

사용자										
<u>ID</u> ↑	Password	이름	성별	핸드폰 번호	플레이리스트 갯수	최대음악재생시간	공통 ID	플레이리스트 등록 ID	플레이리스트 삭제 ID	소유한 이용권

이용권

이용권 이름	이벤트 위	기간	이용서비스	가격

- Relation

- 1. 관리자

- A. 일반적인 attribute는 Table에 그대로 넣었습니다.
- B. 핸드폰번호는 새로운 테이블을 형성하여 관리자의 Primary key(식별ID)를 Foreign key로 가져왔습니다.
- C. 부서와의 관계
 - i. 일하다
 - 1. N쪽으로 1쪽의 Primary key(부서 식별번호)를 Foreign key로 가져왔습니다.
 - ii. 관리하다
 - 1. Total쪽으로 (부서) 관리자의 Primary key를 Foreign key로 가져왔습니다. (부서_관리자 ID)
- D. 음악과의 관계
 - i. 등록
 - 1. N쪽으로 1쪽의 Primary key(음악 ID)를 Foreign key로 가져왔습니다.
 - ii. 삭제
 - 1. N쪽으로 1쪽의 Primary key(음악 ID)를 Foreign key로 가져왔습니다.

- 2. 부서

- A. 부서위치와 관리영역과 같은 multivalued attribute는 새로운 relation을 만들어 primary key(식별번호)를 foreign key로 가져왔습니다.
- B. 관리자와의 관계
 - i. 1번 관리자 설명에서 기술하였습니다.

- 3. 음악

- A. 일반적인 attribute는 Table에 그대로 넣었습니다.
- B. 음악ID라는 primary key attribute를 추가하였습니다.
- C. 관리자와의 관계

- i. 1번 관리자 설명에서 기술 하였습니다.

D. 플레이리스트와의 관계

- i. 삭제

- 1. N:M 관계이기 때문에 플레이리스트_삭제 라는 relation을 생성하고 각각의 relation의 primary key를 foreign key로 가져와서 관계를 명시했습니다.

- ii. 등록

- 1. N:M 관계이기 때문에 플레이리스트_등록 이라는 relation을 생성하고 각각의 relation의 primary key를 foreign key로 가져와서 관계를 명시했습니다.

4. 플레이리스트

- A. 일반적인 attribute는 Table에 그대로 넣었습니다.

- B. 음악과의 관계

- i. 3번 음악 설명에서 기술하였습니다.

- C. 사용자와의 관계

- i. 등록

- 1. N쪽(사용자)으로 1쪽(플레이리스트)의 Primary key(소유한 사용자)를 Foreign key(플레이리스트 등록_ID)로 가져왔습니다.

- ii. 삭제

- 1. N쪽(사용자)으로 1쪽(플레이리스트)의 Primary key(소유한 사용자)를 Foreign key(플레이리스트 삭제_ID)로 가져왔습니다.

5. 사용자

- A. 일반적인 attribute는 Table에 그대로 넣었습니다.

- B. 플레이리스트와의 관계

- i. 4번 플레이리스트 설명에서 기술하였습니다.

- C. 공유와의 관계

- i. Recursive relationship 이기 때문에, 공유한 ID(Foreign key)를 사용자의

Primary key(ID)에 참조하여 가져와 테이블에 추가하였습니다.

D. 이용권과의 관계

- i. 원래 소유의 관계가 사용자:이용권 = 1:N이었으나, 요구사항이 조금 수정되어 N:1로 변경하였습니다.
- ii. 따라서 N쪽(사용자)으로 1쪽(이용권)의 Primary key(이용권 이름)를 Foreign key(소유한 이용권)로 가져왔습니다.

6. 이용권

A. 일반적인 attribute는 Table에 그대로 넣었습니다.

B. 사용자와의 관계

- i. 5번 사용자 설명에서 기술하였습니다.