DB\_Project\_3

­­­

실외, 표지판이(가) 표시된 사진

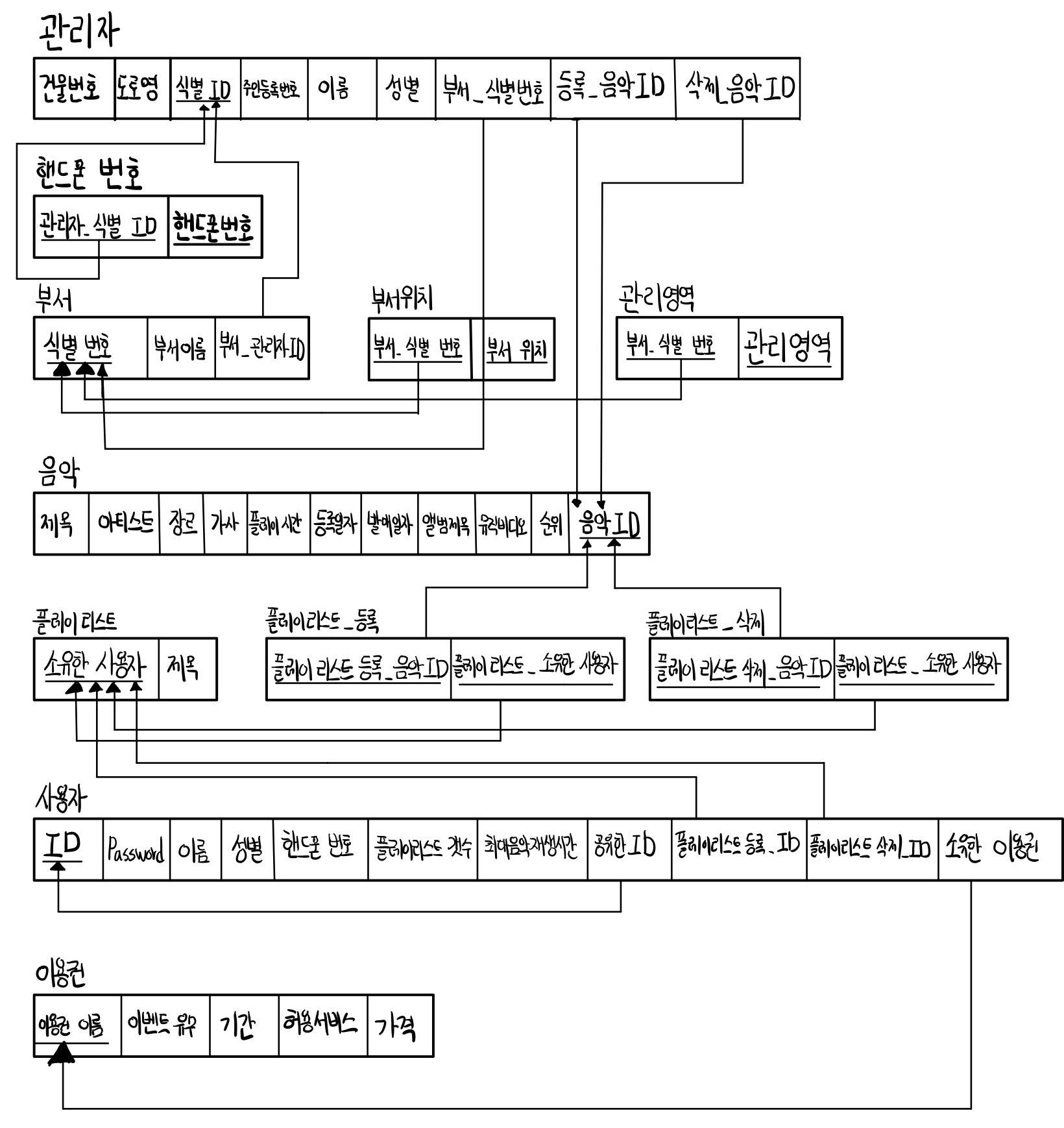
자동 생성된 설명

Datebase Systems

관계형 모델을 이용한 논리적 DB 설계

2016025469

서건식



* Relation

1. 관리자
   1. 일반적인 attribute는 Table에 그대로 넣었습니다.
   2. 핸드폰번호는 새로운 테이블을 형성하여 관리자의 Primary key(식별ID)를 Foreign key로 가져왔습니다.
   3. 부서와의 관계
      1. 일하다
         1. N쪽으로 1쪽의 Primary key(부서 식별번호)를 Foreign key로 가져왔습니다.
      2. 관리하다
         1. Total쪽으로 (부서) 관리자의 Primary key를 Foreign key로 가져왔습니다. (부서\_관리자 ID)
   4. 음악과의 관계
      1. 등록
         1. N쪽으로 1쪽의 Primary key(음악 ID)를 Foreign key로 가져왔습니다.
      2. 삭제
         1. N쪽으로 1쪽의 Primary key(음악 ID)를 Foreign key로 가져왔습니다.
2. 부서
   1. 부서위치와 관리영역과 같은 multivalued attribute는 새로운 relation을 만들어 primary key(식별번호)를 foreign key로 가져왔습니다.
   2. 관리자와의 관계
      1. 1번 관리자 설명에서 기술하였습니다.
3. 음악
   1. 일반적인 attribute는 Table에 그대로 넣었습니다.
   2. 음악ID라는 primary key attribute를 추가하였습니다.
   3. 관리자와의 관계
      1. 1번 관리자 설명에서 기술 하였습니다.
   4. 플레이리스트와의 관계
      1. 삭제
         1. N:M 관계이기 때문에 플레이리스트\_삭제 라는 relation을 생성하고 각각의 relation의 primary key를 foreign key로 가져와서 관계를 명시했습니다.
      2. 등록
         1. N:M 관계이기 때문에 플레이리스트\_등록 이라는 relation을 생성하고 각각의 relation의 primary key를 foreign key로 가져와서 관계를 명시했습니다.
4. 플레이리스트
   1. 일반적인 attribute는 Table에 그대로 넣었습니다.
   2. 음악과의 관계
      1. 3번 음악 설명에서 기술하였습니다.
   3. 사용자와의 관계
      1. 등록
         1. N쪽(사용자)으로 1쪽(플레이리스트)의 Primary key(소유한 사용자)를 Foreign key(플레이리스트 등록\_ID)로 가져왔습니다.
      2. 삭제
         1. N쪽(사용자)으로 1쪽(플레이리스트)의 Primary key(소유한 사용자)를 Foreign key(플레이리스트 삭제\_ID)로 가져왔습니다.
5. 사용자
   1. 일반적인 attribute는 Table에 그대로 넣었습니다.
   2. 플레이리스트와의 관계
      1. 4번 플레이리스트 설명에서 기술하였습니다.
   3. 공유와의 관계
      1. Recursive relationship 이기 때문에, 공유한 ID(Foreign key)를 사용자의 Primary key(ID)에 참조하여 가져와 테이블에 추가하였습니다.
   4. 이용권과의 관계
      1. 원래 소유의 관계가 사용자:이용권 = 1:N이었으나, 요구사항이 조금 수정되어 N:1로 변경하였습니다.
      2. 따라서 N쪽(사용자)으로 1쪽(이용권)의 Primary key(이용권 이름)를 Foreign key(소유한 이용권)로 가져왔습니다.
6. 이용권
   1. 일반적인 attribute는 Table에 그대로 넣었습니다.
   2. 사용자와의 관계
      1. 5번 사용자 설명에서 기술하였습니다.