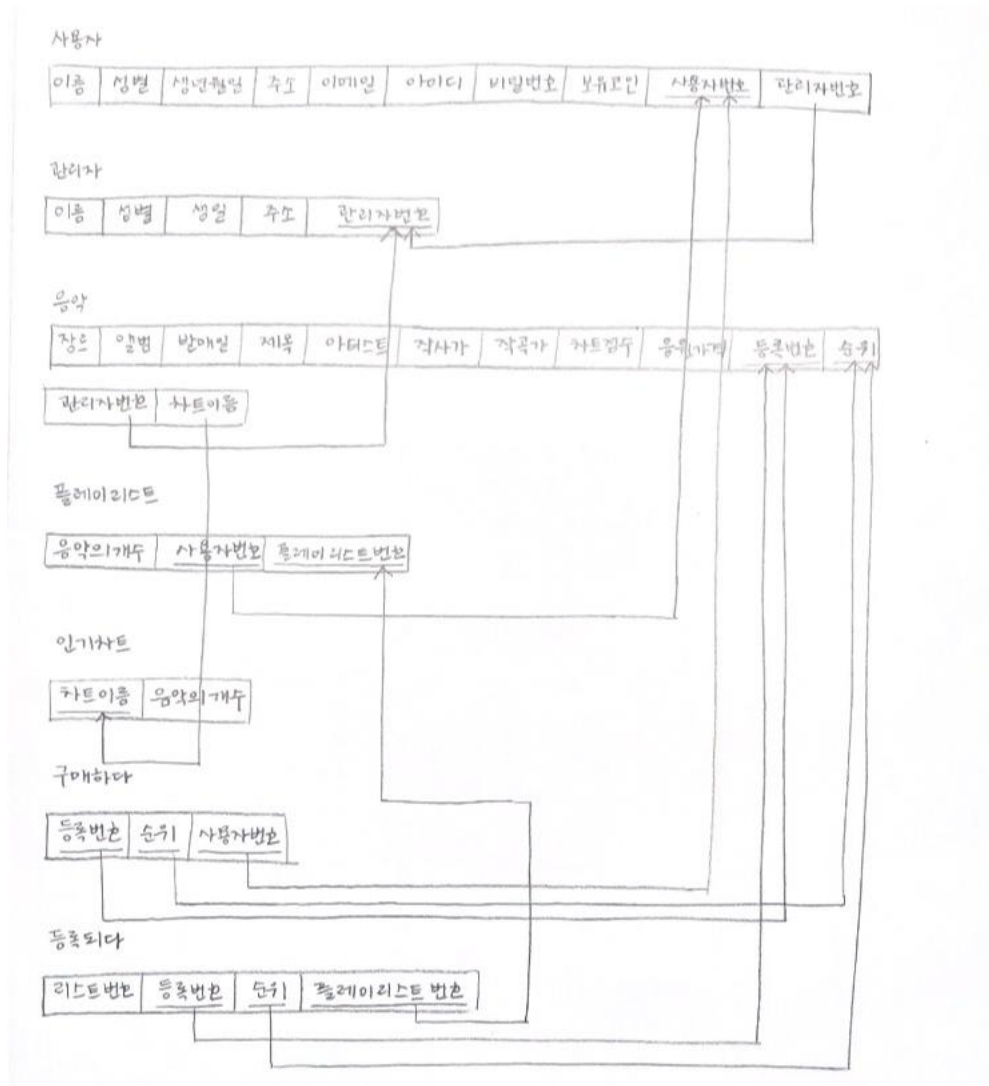


<프로젝트 3: 관계형 스키마를 이용한 논리적 (logical) DB 설계>

컴퓨터소프트웨어학부 2018008331 박민경



수정사항

1. 플레이리스트의 key attribute로 플레이리스트번호를 추가하였다.
2. 관리자의 attribute인 주민번호를 삭제하였다.
3. 인기차트의 key attribute로 차트이름을 추가하였다.

#사용자

이름	성별	생년월일	주소	이메일	아이디	비밀번호	보유코인	<u>사용자번호</u>	관리자번호
----	----	------	----	-----	-----	------	------	--------------	-------

[사용자]는 regular entity type이다.

attribute: entity type의 모든 simple attribute을 가진다. 그리고 관리자의 PK를 FK로 가진다.

primary key: entity type의 key attribute가 유일하므로, 사용자번호를 primary key로 가진다.

<사용자_관리하다> 는 1:N relationship type이며, 이것을 표현하기 위해 [관리자]의 primary key인 관리자번호를 N-side인 [사용자]의 foreign key로 가져온다.

#관리자

이름	성별	생일	주소	<u>관리자번호</u>
----	----	----	----	--------------

[관리자]는 regular entity type이다.

attribute: entity type의 모든 simple attribute을 가진다.

primary key: entity type의 key attribute가 유일하므로, 관리자번호를 primary key로 가진다.

#음악

장르	앨범	발매일	제목	아티스트	작사가	작곡가	차트점수	음원가격	<u>등록번호</u>
<u>순위</u>	관리자번호	<u>차트이름</u>							

[음악]은 regular entity type이다.

attribute: entity type의 모든 simple attribute을 가진다. 그리고 관리자의 PK를 FK로 가지고, 인기 차트의 PK를 FK로 가진다.

primary key: entity type의 key attribute인 등록번호와 순위 중, 등록번호를 primary key로 가진다.

<음악_관리하다> 는 1:N relationship type이며, 이것을 표현하기 위해 [관리자] entity type의 primary key인 관리자번호를 N-side인 [음악]의 foreign key로 가져온다.

<진입하다> 는 1:N relationship type이며, 이것을 표현하기 위해 [인기차트] entity type의 primary key인 차트이름을 N-side인 [음악]의 foreign key로 가져온다.

#플레이리스트

음악의 개수	<u>사용자번호</u>	<u>플레이리스트번호</u>
--------	--------------	-----------------

[플레이리스트]는 weak entity type이다.

attribute: entity type의 모든 simple attribute을 가진다. 그리고 owner인 [사용자]의 PK인 사용자번호를 FK로 가진다.

primary key: entity type의 key attribute인 플레이리스트번호와 FK로 가져온 사용자번호를 결합하여 primary key로 가진다.

<수정하다> 는 1:1 relationship type이며, 이것을 표현하기 위해 [사용자] entity type의 primary key인 사용자번호를 [플레이리스트]의 foreign key로 가져온다.

#인기차트

<u>차트이름</u>	음악의 개수
-------------	--------

[인기차트]는 regular entity type이다.

attribute: entity type의 모든 simple attribute을 가진다.

primary key: entity type의 key attribute가 유일하므로, 차트이름을 primary key로 가진다.

#구매하다

<u>등록번호</u>	<u>순위</u>	<u>사용자번호</u>
-------------	-----------	--------------

[구매하다]는 binary M:N relationship type이다.

attribute: relationship type의 simple attribute은 없고, participating entity type인 [음악]의 PK(등록번호, 순위)와 [사용자]의 PK(사용자번호)를 FK로 가진다.

primary key: 모든 FK (등록번호, 순위, 사용자번호)을 결합하여 primary key로 가진다.

#등록되다

리스트번호	<u>등록번호</u>	<u>순위</u>	<u>플레이리스트번호</u>
-------	-------------	-----------	-----------------

[등록되다]는 binary M:N relationship type이다.

attribute: relationship type의 simple attribute인 리스트번호를 가지고, participation entity type인

[음악]의 PK(등록번호, 순위)와 [플레이리스트]의 PK(플레이리스트번호)를 FK로 가진다.

primary key: 모든 FK (등록번호, 순위, 플레이리스트번호)를 결합하여 primary key로 가진다.