

데이터베이스

강의 노트

제 12 회차
논리적 설계 단계

❖ 학습목표

- 릴레이션 스키마 생성 단계를 나열 할 수 있다.
- 개체 변환 방법에 대해 설명할 수 있다.
- 관계 변환 방법에 대해 설명할 수 있다.
- 릴레이션을 단순화시킬 수 있다.
- 무결성 제약조건의 정의 기준을 설명할 수 있다.

❖ 학습내용

- 릴레이션 스키마 생성 데이터 독립성에 대해 설명할 수 있다.
- 릴레이션 단순화 방법과 릴레이션 생성 사례

릴레이션 스키마 생성

1. 논리적 설계 개요
2. 릴레이션 스키마 생성 방법
3. 개체 변환
4. 관계 변환
5. 다중치 속성 변환

1. 논리적 설계 개요

1) 논리적 설계 개요

논리적 설계(Logical Design)란?

개념적 데이터 모델을 DBMS가 지원하는 논리적 데이터 모델로 변환하는 것

논리적 설계의 주요 업무

논리적 모델링

ERD로 표현된 개념적 스키마를 **논리적 스키마(릴레이션 스키마)**로 변환한 다음, **정규화*** 과정을 통해서 보다 바람직한 모델로 변환하고, 요구 분석 명세서를 기초로 **무결성 제약조건을 정의**

***정규화 과정** : 어떤 정보도 잃지 않으면서 주어진 릴레이션들을 보다 바람직한 형태로 변환해가는 과정

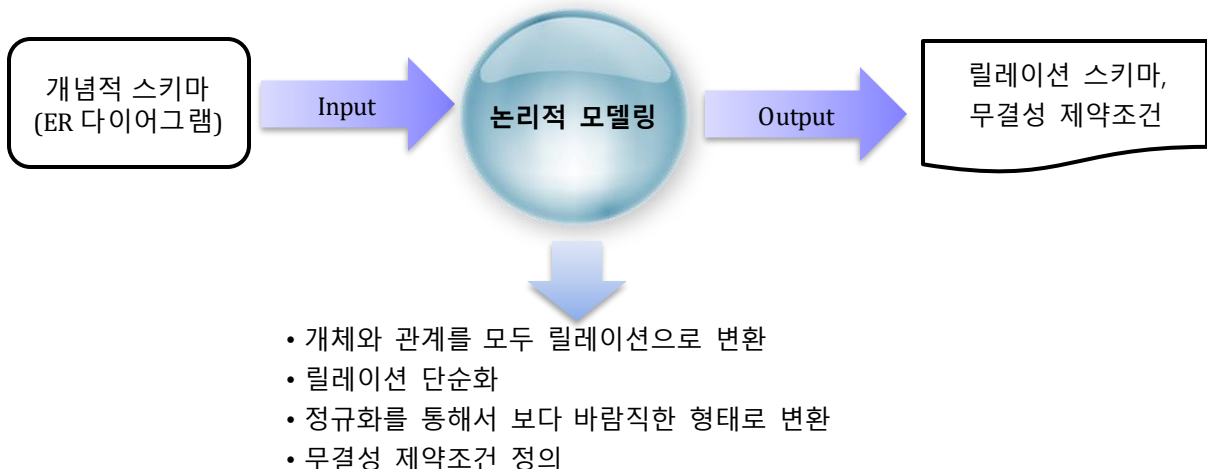
트랜잭션 인터페이스 설계

- 트랜잭션 모델링을 기초로 인터페이스를 설계

스키마 평가 및 정제

- 논리적 스키마를 정량적 정보와 성능 평가기준에 따라 평가해서 정제

2) 논리적 모델링의 입출력



2. 릴레이션 스키마 생성

1) 릴레이션 스키마 생성

릴레이션 스키마(Relation Schema)란?

- 가장 대표적인 논리적 데이터 모델인 관계 데이터 모델의 기본이 되는 릴레이션을 구성하는 속성들의 집합
- 릴레이션 이름과 속성 이름들로 표현

2) 릴레이션 스키마 생성 단계

개체 릴레이션:
개체명(속성1, 속성2, ...)

관계 릴레이션:
관계명(속성1, 속성2, ...)

ERD의 개체와 관계를 릴레이션 스키마로 변환
: 릴레이션명(속성1, 속성2, ...)

릴레이션 스키마에 기본 키 표시
: 기본 키 아래 **밑줄** 표시

릴레이션 스키마의 단순화
: **일 대 일**, **일 대 다** 관계 유형 단순화

정규화(Normalization)
: 보다 바람직한 릴레이션 형태로 변환

3. 개체 변환

1) 단순 속성을 갖는 개체 변환 방법

변환 방법

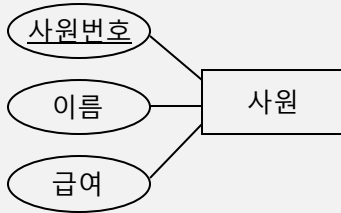
모든 속성이 릴레이션의 속성이 됨

기본 키

개체의 후보 키 가운데 하나가 릴레이션의 기본 키가 됨

변환 예

단순 속성을 갖는 개체



개체 변환

릴레이션 스키마

직원(직원번호, 이름, 급여)

리스트 형태 표현

직원

직원번호	이름	급여

테이블 형태 표현

3. 개체 변환

2) 복합 속성을 갖는 개체 변환 방법

변환 방법

복합 속성을 구성하는 **모든** 속성이 릴레이션의 속성이 됨

기본 키

복합 속성이 일반 속성인 경우와 후보 키에 대응하는 속성인 경우를 구분해서 판단

복합 속성이 일반 속성인 경우

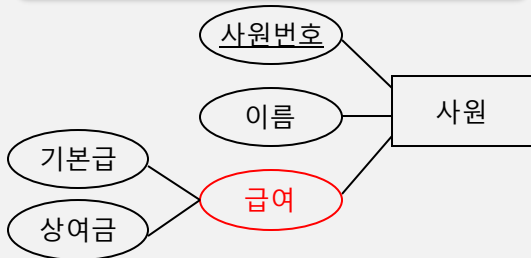
기본 키

개체의 후보 키 가운데 하나가 릴레이션의 기본 키가 됨

변환 예

일반 속성인 복합 속성을 갖는 개체

릴레이션 스키마



개체 변환

직원(직원번호, 이름, 기본급, 상여금)

후보 키 속성이 복합 속성인 경우

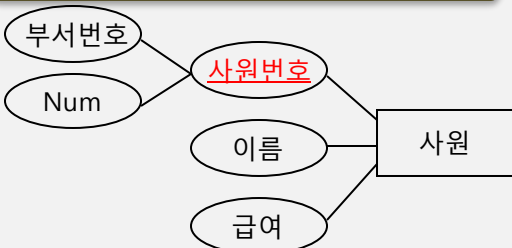
기본 키

복합 속성을 구성하는 **모든** 속성들이 기본 키가 됨

변환 예

후보 키가 복합 속성인 개체

릴레이션 스키마



개체 변환

직원(부서번호, Num, 이름, 급여)

3. 개체 변환

2) 복합 속성을 갖는 개체 변환 방법



밑줄 연결에 대한 비교

[비교] 다음 두 릴레이션 스키마는 다른 의미를 가짐

사원(부서번호, Num, 이름, 기본급, 상여금)

사원(부서번호, Num, 이름, 기본급, 상여금)

- 전자는 **부서번호와 Num이라는 2개의 복합 속성이 기본 키**임을 의미함
- 후자는 **부서번호나 Num 가운데 하나가 기본 키**가 될 수 있음을 의미함

3. 개체 변환

3) 약한 개체 변환 방법

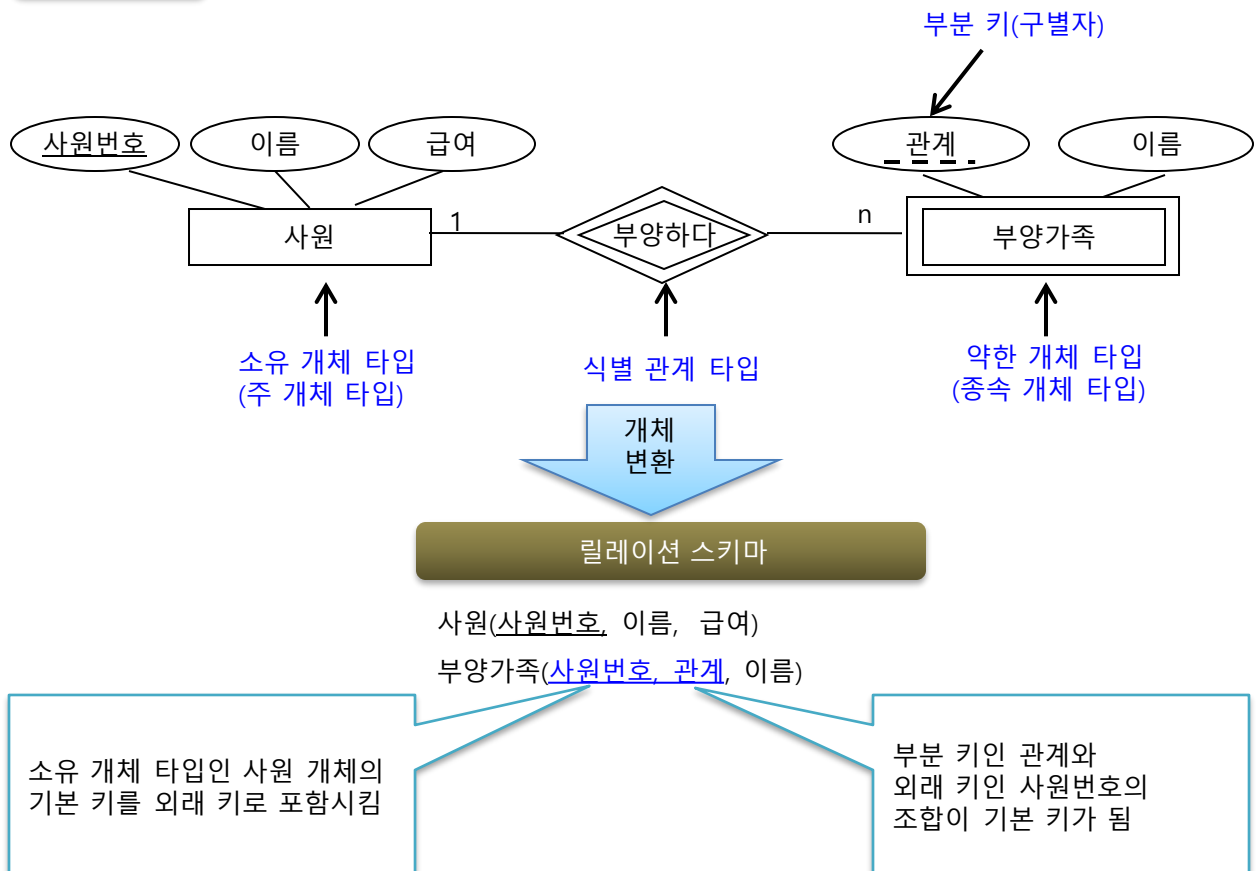
변환 방법

소유 개체(주 개체) 타입을 갖는 약한 개체 타입에 대해서 릴레이션을 생성하고,
약한 개체 타입에 속한 모든 단순 속성들을 릴레이션에 포함시키고,
소유 개체 타입의 기본 키를 약한 개체 타입의 외래 키로 포함시킴

기본 키

약한 개체 타입에 해당하는 릴레이션의 기본 키는 약한 개체 타입의 **부분 키(구별자)**와
소유 개체 타입에 해당하는 릴레이션을 참조하는 **외래 키의 조합**이 기본 키가 됨

변환 예



4. 관계 변환

1) 관계 변환 특징 및 기본 키

특징

"관계 자체의 속성" 및 "관계에 속하는 개체들의 기본 키 속성"이 모두 릴레이션의 속성이 됨

기본 키

관계의 유형에 따라 기본 키를 결정하는 방법에 차이가 있음



2) 일 대 일(1 : 1) 관계 변환 방법

변환 방법

관계에 속하는 개체들의 기본 키 속성과 관계 자체의 속성(있는 경우)을 관계 릴레이션의 속성으로 표현



이 때, 개체의 기본 키는 관계 릴레이션의 **외래 키**가 됨

기본 키

관계에 포함된 양쪽 개체의 **기본 키 가운데 하나**가 기본 키가 될 수 있음

주의사항

기본 키의 선택은 업무의 성격에 맞게 결정해야 함

4. 관계 변환

2) 일 대 일(1 : 1) 관계 변환 방법



일 대 일 관계의 기본 키 설정 기준

다음 2가지 형태의 릴레이션 스키마로 변환이 가능한데, 어떤 것을 선택하는 것이 좋을까요?

릴레이션 스키마 { 관리하다(사원번호(FK), 부서번호(FK), 임명일자) --- ①
관리하다(사원번호(FK), 부서번호(FK), 임명일자) --- ②

🍊 업무의 성격을 고려해서 기본 키를 선택한다. 🍊

- 1) 업무를 처리함에 있어서 부서번호로 부서장을 확인하는 경우가 많다면, 부서번호를 기본 키로 정의함
- 2) 업무를 처리함에 있어서 어떤 사원이 어느 부서의 부서장을 맡고 있는지 확인하는 경우가 많다면, 사원번호를 기본 키로 정의함

‘관리하다’ 릴레이션

사원번호	부서번호	임명일자
1234	100	03/01/2010
2553	200	03/01/2010
3612	300	09/01/2011
4155	400	01/01/2012
		:

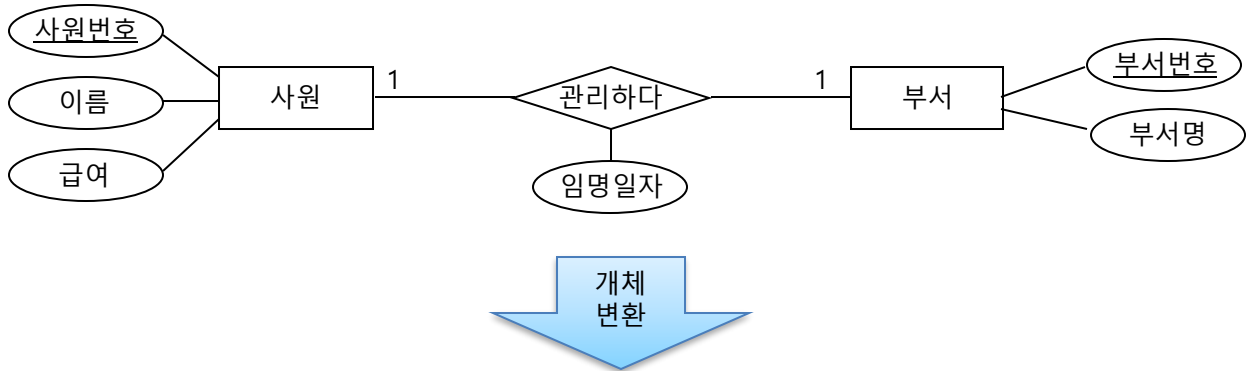
사원번호와 부서번호 모두 유일한 값을 가지므로 둘 다 기본 키가 될 수 있지만,
업무를 처리함에 있어서 부서번호로 부서장을 확인하는 경우가 많다면
사원번호 보다는 **부서번호를 기본 키로 정의**하는 것이 바람직함

4. 관계 변환

2) 일 대 일(1 : 1) 관계 변환 방법

변환 예

일 대 일 (1 : 1) 관계 유형



릴레이션 스키마

{ 관리하다(사원번호(FK), 부서번호(FK), 임명일자)
 { 관리하다(사원번호(FK), 부서번호(FK), 임명일자)

혹은

기본 키는 밑줄로, 외래 키는
 FK(Foreign Key)로 구분해서
 표시

4. 관계 변환

3) 일 대 다 (1 : n) 관계 변환 방법

변환 방법

관계에 속하는 개체들의 기본 키 속성과 관계 자체의 속성(있는 경우)을 릴레이션의 속성으로 표현

기본 키

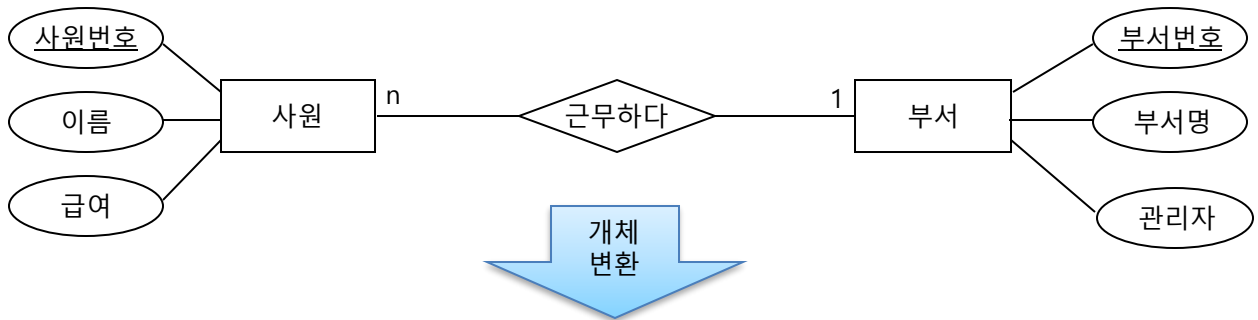
관계에 포함된 개체 가운데 **n**측 개체의 기본 키가 기본 키

주의사항

1측 개체의 기본 키는 관계 릴레이션에서 중복되므로,
n 측 개체의 기본 키가 관계 릴레이션의 기본 키가 되어야 함

변환 예

일 대 다(1 : n) 관계 유형



릴레이션 스키마

근무하다(사원번호(FK), 부서번호(FK))

어떤 사원이 어떤 부서에서 근무하는지를 표현하는 '근무하다'라는 관계의 경우, 많은 사원이 같은 부서에 근무하게 되므로 동일한 부서번호가 중복되고, 따라서 부서번호는 기본 키가 될 수 없음

4. 관계 변환

3) 일 대 다 (1 : n) 관계 변환 방법



여기서 잠깐!

‘근무하다’ 릴레이션의 데이터(릴레이션 인스턴스)를 확인하면, 부서번호가 기본 키가 될 수 없는 이유를 명확히 알 수 있음

‘근무하다’ 릴레이션

사원번호	부서번호
11135	100
11233	100
12336	100
12476	200
:	:

사원번호 11135, 11233, 12336이 모두 100번 부서에 근무하기 때문에 동일한 부서번호가 중복되어 있으므로 부서번호는 기본 키가 될 수 없음

4. 관계 변환

4) 다 대 다 ($m : n$) 관계 변환 방법

변환 방법

관계에 속하는 개체들의 기본 키 속성과 관계 자체의 속성(있는 경우)을 릴레이션의 속성으로 표현

기본 키

관계에 포함된 양쪽 개체의 기본 키들을 구성하는 모든 속성들의 조합(Combination)이 기본 키

주의사항

m 측 개체의 기본 키와 n 측 개체의 기본 키는 각각 관계 릴레이션에서 중복될 수 있으므로, 이들 기본 키가 조합되어 관계 릴레이션의 기본 키가 되어야 함

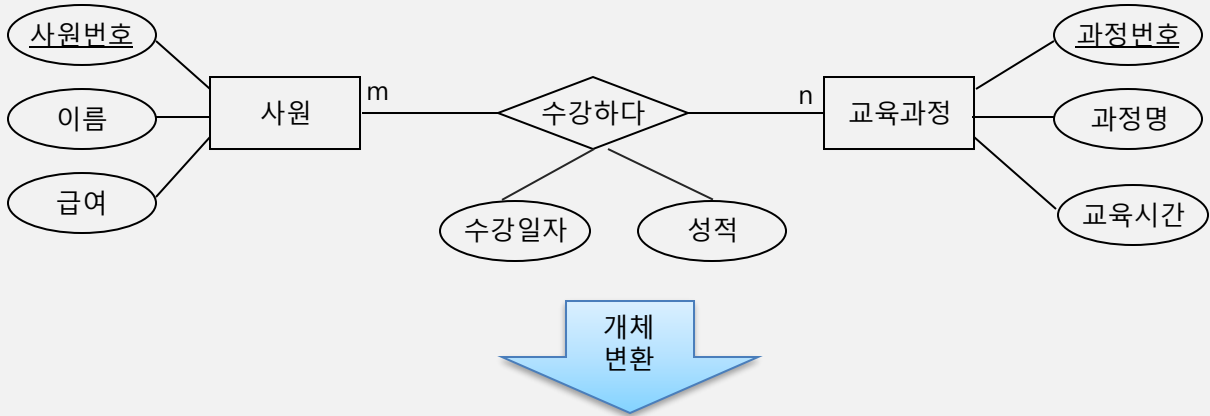
4. 관계 변환

4) 다 대 다 (m : n) 관계 변환 방법

변환 예 1

다 대 다(m : n) 관계 유형

여러 명의 사원이 여러 교육과정을 수강하는 관계 표현



개체
변환

릴레이션 스키마

수강하다(사원번호(FK), 과정번호(FK), 수강일자, 성적)



여기서 잠깐!

'수강하다' 릴레이션의 데이터(릴레이션 인스턴스)를 확인하면, 사원번호나 과정번호 하나만으로는 기본 키가 될 수 없는 이유를 명확히 알 수 있음

'수강하다' 릴레이션

사원번호	과정번호	수강일자	성적
11135	100	02/10/2013	A
11135	200	03/15/2013	B
11236	100	02/10/2013	B
12476	200	03/15/2013	A
:	:	:	:

각각이 중복되어 있으므로 사원번호나 과정번호 어느 한 속성만으로는 기본 키가 될 수 없음

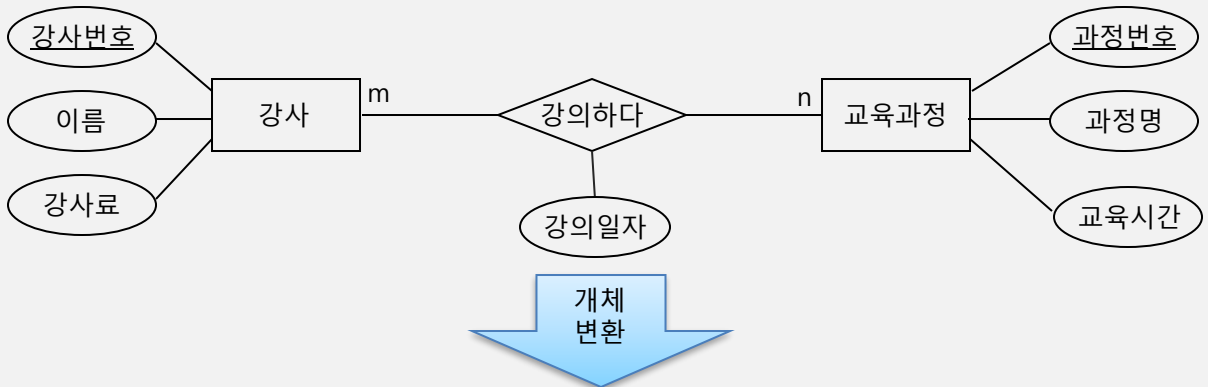
4. 관계 변환

4) 다 대 다 (m : n) 관계 변환 방법

변환 예 2

다 대 다(m:n) 관계 유형

여러 명의 강사가 여러 교육과정을 강의하는 관계 표현



릴레이션 스키마

강의하다(강사번호(FK), 과정번호(FK), 강의일자)

다 대 다 유형이므로 원래 사원번호와 과정번호가 조합되어 기본 키가 되어야 하지만, 같은 교육과정을 같은 강사가 다른 날짜에 여러 번 강의할 수 있기 때문에 이 두 가지 속성만으로 기본 키가 정의되면, '강의일자' 속성의 값만 다른 경우 데이터를 삽입할 수 없게 됨. 따라서, 기본 키에 '강의일자' 속성을 포함시켜야 함



여기서 잠깐!

'강의하다' 릴레이션의 데이터(릴레이션 인스턴스)를 확인하면, 강사번호와 과정번호의 조합만으로는 기본 키가 될 수 없는 이유를 명확히 알 수 있음

'강의하다' 릴레이션

강사번호	과정번호	강의일자
4572	100	02/10/2013
2523	200	03/15/2013
4572	100	04/10/2013
4572	300	05/15/2013
:	:	:

같은 강사가 같은 강의를 다른 날짜에 여러 번 강의할 수 있는데, 만약 (강사번호, 과정번호)를 기본 키로 설정하면 두 번째 강의한 데이터는 삽입할 수 없게 됨

4. 관계 변환

5) 순환 관계 변환

일 대 다 (1:n) 순환 관계 변환 방법

변환 방법

- 관계 릴레이션을 별도로 생성하지 않고, 개체의 기본 키를 외래 키로 포함시킴
- 만약 관계 자체의 속성이 있으면, 그 속성도 개체 릴레이션에 포함시킴

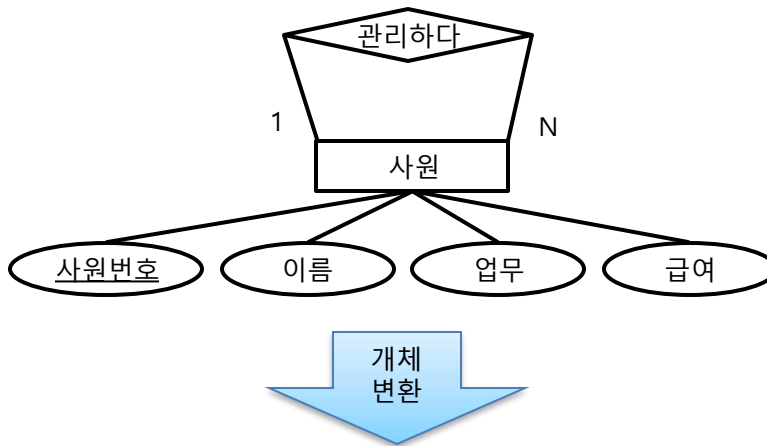
기본 키

개체의 기존 기본 키가 그대로 기본 키

변환 예

한 명의 사원(부서장)이 여러 사원을 관리하는 관계 표현

일 대 다 순환 관계



릴레이션 스키마

사원(사원번호, 이름, 업무, 급여, **관리자사원번호(FK)**)

관리하다'라는 순환 관계를 표현하기 위해 추가된 외래 키

4. 관계 변환

5) 순환 관계 변환

다 대 다(m : n) 순환 관계 변환 방법

변환 방법

- 관계 릴레이션을 별도로 생성하고, 개체의 기본 키를 2개의 외래 키로 포함시킴
- 만약 관계 자체의 속성이 있으면, 그 속성도 개체 릴레이션에 포함시킴

기본 키

2개의 외래 키가 조합되어 관계 릴레이션의 기본 키

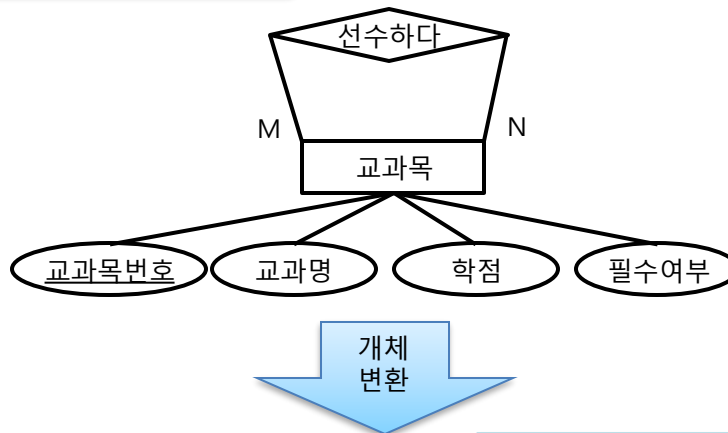
주의사항

동일한 기본 키가 2개 추가되므로, 2개의 속성 이름이 서로 달라야 함

변환 예

많은 교과목이 여러 다른 교과목의 선수 과목이 되는 관계 표현

다 대 다 순환 관계



릴레이션 스키마

교과목(교과목번호, 교과명, 학점, 필수여부)

선수하다(교과목번호(FK), 선수교과목번호(FK))

어떤 교과목의 선수과목을 표현하기 위해서, 교과목 개체의 기본 키(교과목번호)가 관계 릴레이션에 2개의 외래 키(교과목번호, 선수교과목번호)로 포함되고, 이 2개의 외래 키가 조합되어 기본 키가 됨

4. 관계 변환

6) 3진 이상 관계 변환

변환 방법

- 3진 이상의 관계 타입에 대해서 **별도의 릴레이션을 생성**
- 관계 타입에 참여하는 **모든 개체 타입의 기본 키**를 관계 릴레이션에 **외래 키**로 포함
- 관계 타입이 갖고 있는 모든 단순 속성(복합 속성을 갖고 있는 경우, 복합 속성을 구성하는 단순 속성들)을 릴레이션에 포함

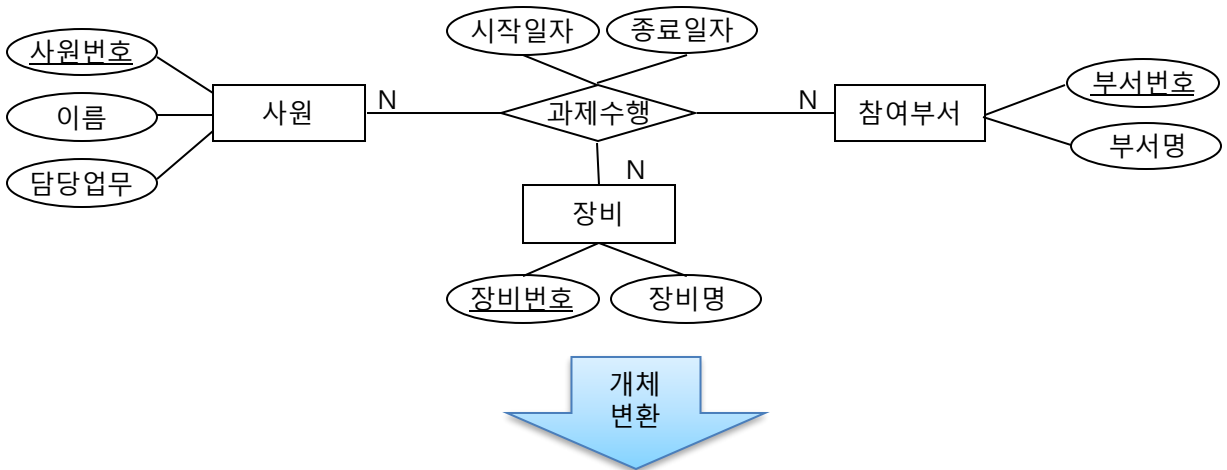
기본 키

일반적으로 **외래 키들의 조합**이 관계 릴레이션의 기본 키

변환 예

하나의 과제(프로젝트)에 많은 사원과 부서, 그리고 많은 장비가 연관됨을 표현

3진 이상 관계



릴레이션 스키마

과제수행(사원번호, 부서번호, 장비번호, 시작일자, 종료일자)

5. 다중치 속성 변환

변환 방법

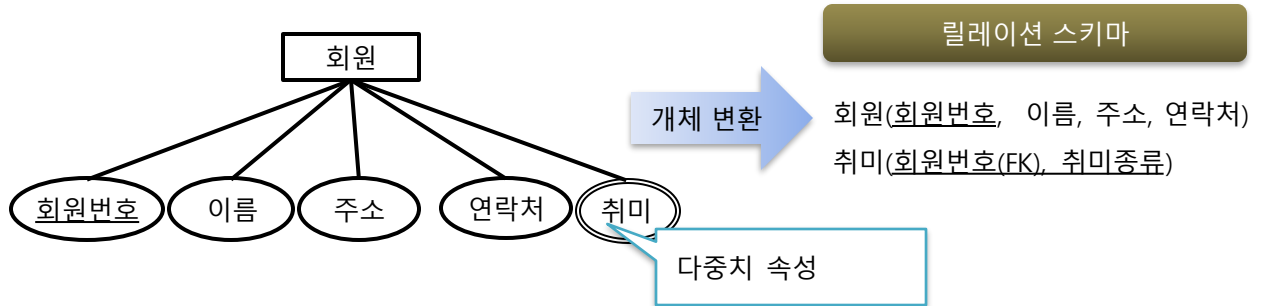
- 다중치 속성을 위한 **별도의 릴레이션을 생성**
- 다중치 속성을 갖는 개체 타입이나 관계 타입에 해당하는 릴레이션의 **기본 키를 외래 키로 포함**

기본 키

외래 키와 다중치 속성의 조합이 기본 키

변환 예

하나 이상 여러 개의 취미 정보를 포함하는 회원 개체 표현



릴레이션 단순화 방법과 릴레이션 생성 사례

1. 릴레이션 단순화
2. 릴레이션 생성 사례
3. 무결성 제약조건 정의

1. 릴레이션 단순화

1) 일 대 일(1 : 1) 관계 단순화

관계를 표현하는 릴레이션을 따로 생성하지 않고, 관계에 포함된 개체 가운데 한 개체에 관계의 속성을 모두 포함시켜서 전체 릴레이션을 단순화시킬 수 있음

단순화 방법

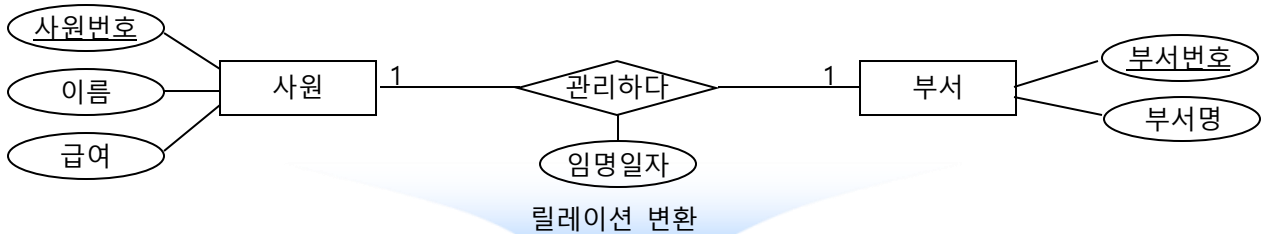
관계에 포함된 두 개체 릴레이션 가운데 하나에 다음을 추가

관계에 포함된 다른 개체의 기본 키에 대응하는 속성

관계 자체의 속성 (있는 경우)

기존의 관계 릴레이션은 삭제

단순화 예



릴레이션 변환

직원(직원번호, 이름, 급여)

부서(부서번호, 부서명)

관리하다(부서번호(FK), 직원번호(FK), 임명일자) **혹은** 관리하다(직원번호(FK), 부서번호(FK), 임명일자)

릴레이션 단순화

부서(부서번호, 부서명, **부서장번호(FK)**, 임명일자)
직원(직원번호, 이름, 급여)

혹은

직원(직원번호, 이름, **관리부서번호(FK)**, 임명일자)
부서(부서번호, 부서명)

- 부서 개체 릴레이션에 직원 개체의 기본 키(직원번호)와 관계 자체의 속성을 추가함
- '관리하다' 릴레이션은 삭제함

- 직원 개체 릴레이션에 부서 개체의 기본 키(부서번호)와 관계 자체의 속성을 추가함
- '관리하다' 릴레이션은 삭제함

1. 릴레이션 단순화

1) 일 대 일(1 : 1) 관계 단순화



여기서 잠깐!

‘관리하다’ 릴레이션을 단순화시킨 아래 2가지 가운데 어떤 것을 선택하는 것이 좋을까요?

a) 부서 개체 릴레이션에 관계를 포함시킴

부서(부서번호, 부서명, **부서장번호(FK)**, **임명일자**)
 사원(사원번호, 이름, 급여)

혹은

b) 사원 개체 릴레이션에 관계를 포함시킴

사원(사원번호, 이름, 급여, **관리부서번호(FK)**, **임명일자**)
 부서(부서번호, 부서명)



a)의 경우, 모든 부서에 부서장이 있으므로 부서 개체 릴레이션에 부서장번호(사원 개체의 기본 키인 사원번호에 해당함)를 포함시키는 것이 타당함



b)의 경우, 사원 가운데 몇 명만이 부서장이지만 모든 사원 개체에 ‘관리부서번호’라는 속성 (부서 개체의 기본 키인 부서번호에 해당함)이 포함되어 대부분 널(Null) 값을 갖게 되므로 기억장소가 낭비됨

⇒ 두 가지 경우 각각 릴레이션에 포함된 데이터(릴레이션 인스턴스)를 확인하면 어떤 개체에 관계를 포함시키는 것이 좋은지 명확히 알 수 있음

1. 릴레이션 단순화

1) 일 대 일(1 : 1) 관계 단순화



여기서 잠깐!

a) 부서 개체에 관계를 포함시킨 경우

부서 릴레이션

부서번호	부서명	부서장번호	임명일자
100	관리부	1234	03/01/2010
200	경리부	2553	03/01/2010
300	개발부	3612	09/01/2011
:	:	:	:

부서장번호와 임명일자 속성이
널 값을 갖는 데이터가 없음

b) 사원 개체에 관계를 포함시킨 경우

사원 릴레이션

사원번호	이름	급여	관리부서번호	임명일자
1234	박동희	4500000	100	03/01/2010
1326	김철수	2750000		
1524	이영희	1700000		
2553	김창수	4700000	200	03/01/2010
3612	민동규	6500000	300	09/01/2011
:	:	:	:	:

부서장이 아닌 사원은 모두 관리부서번호와 임명일자
속성이 널 값을 갖게 되므로 기억장소가 낭비됨

1. 릴레이션 단순화

2) 일 대 다 (1 : n) 관계 단순화

관계를 표현하는 릴레이션을 따로 생성하지 않고, 관계에 포함된 개체 가운데 **n측 개체**에 관계의 속성을 모두 포함시켜서 전체 릴레이션을 단순화시킬 수 있음

단순화 방법

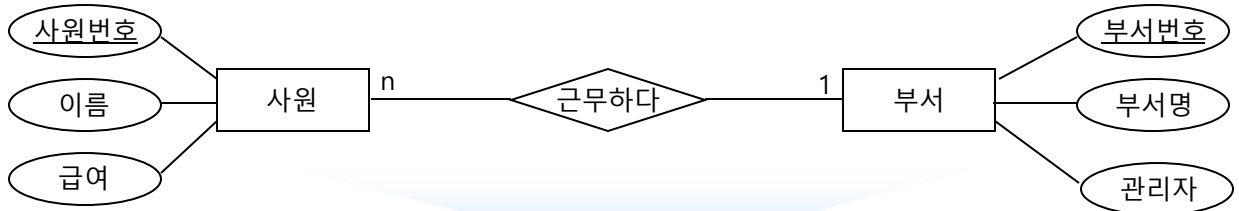
관계의 n측에 있는
개체 릴레이션에 다음을 추가

관계의 1측에 있는 개체의 기본 키에 대응하는 속성

관계 자체의 속성 (있는 경우)

기존의 관계 릴레이션은 삭제

단순화 예



릴레이션 변환

직원(직원번호, 이름, 급여)
부서(부서번호, 부서명)
근무하다(직원번호(FK), 부서번호(FK))

릴레이션 단순화

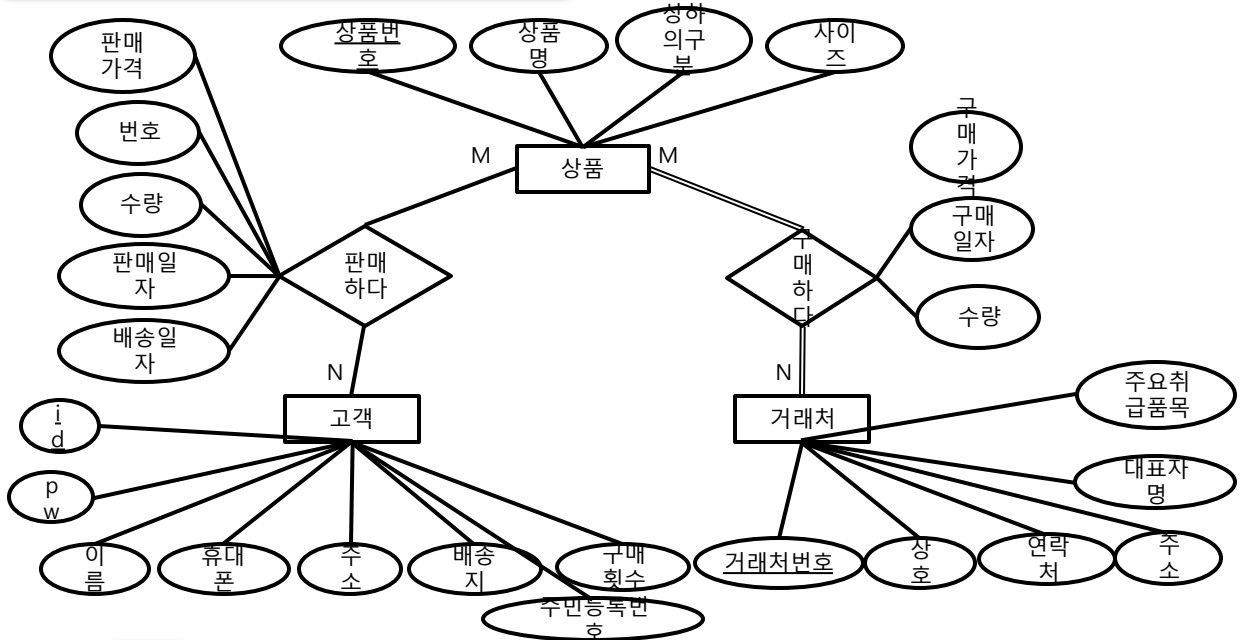
직원(직원번호, 이름, **부서번호(FK)**)
부서(부서번호, 부서명, 관리자)

n측 개체인 직원 개체에 1측 개체인 부서
개체의 기본 키를 외래 키로 포함시킴

2. 릴레이션 생성 사례

1) 온라인 쇼핑몰의 릴레이션 스키마 작성

온라인 쇼핑몰 DB 구축을 위한 ERD



개체 릴레이션 생성 및
기본 키 설정

상품(상품번호, 상품명, 상하의구분, 사이즈)

고객(id, pw, 주민등록번호, 이름, 휴대폰, 주소, 배송지, 구매횟수)

거래처(거래처번호, 상호, 연락처, 주소, 대표자명, 주요취급품목)

관계 릴레이션 생성 및
기본 키 설정

판매하다(상품번호(FK), 고객id(FK), 번호, 판매가격, 수량, 판매일자, 배송일자)

구매하다(상품번호(FK), 거래처번호(FK), 구매가격, 수량, 구매일자)

릴레이션 스키마
단순화

2개의 관계 릴레이션이 모두 다 대 다(M:N) 유형이므로 단순화시킬 수 없음

최종 릴레이션 스키마

상품(상품번호, 상품명, 상하의구분, 사이즈)

고객(id, pw, 주민등록번호, 이름, 휴대폰, 주소, 배송지, 구매횟수)

거래처(거래처번호, 상호, 연락처, 주소, 대표자명, 주요취급품목)

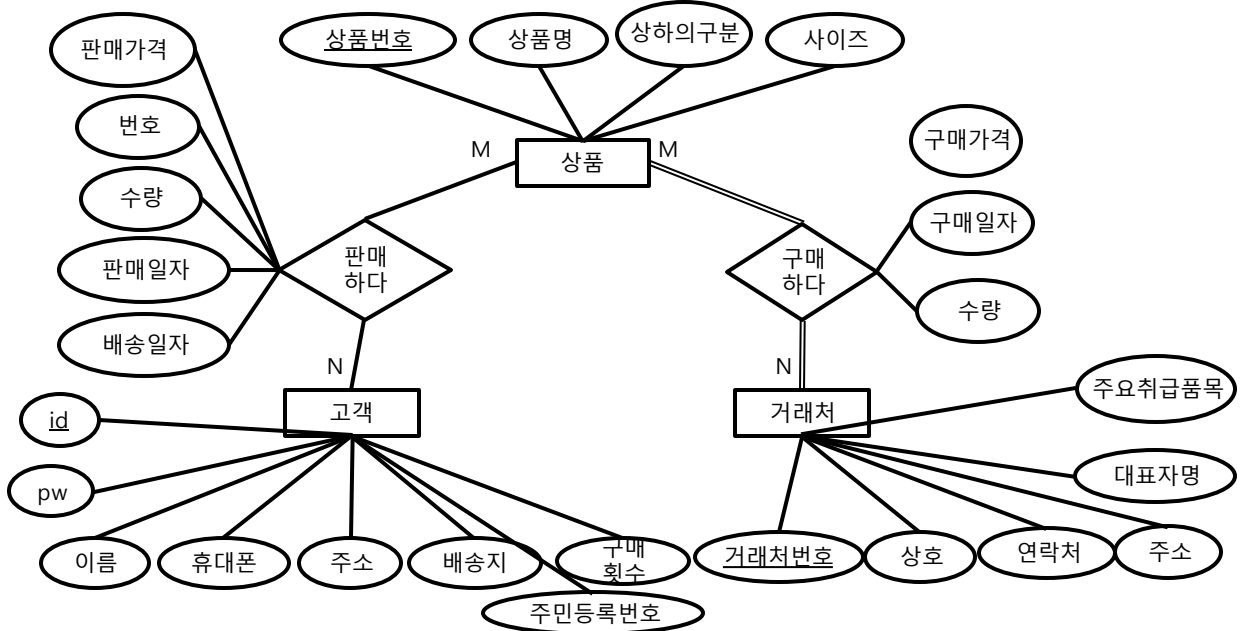
판매하다(상품번호(FK), 고객id(FK), 번호, 판매가격, 수량, 판매일자, 배송일자)

구매하다(상품번호(FK), 거래처번호(FK), 구매가격, 수량, 구매일자)

2. 릴레이션 생성 사례

1) 온라인 쇼핑몰의 릴레이션 스키마 작성

온라인 쇼핑몰 DB 구축을 위한 ERD



개체 릴레이션 생성 및
기본 키 설정

상품(상품번호, 상품명, 상하의구분, 사이즈)
고객(id, pw, 주민등록번호, 이름, 휴대폰, 주소, 배송지, 구매횟수)
거래처(거래처번호, 상호, 연락처, 주소, 대표자명, 주요취급품목)

관계 릴레이션 생성 및
기본 키 설정

판매하다(상품번호(FK), 고객id(FK), 번호, 판매가격, 수량, 판매일자, 배송일자)
구매하다(상품번호(FK), 거래처번호(FK), 구매가격, 수량, 구매일자)

릴레이션 스키마
단순화

2개의 관계 릴레이션이 모두 다 대 다(M:N) 유형이므로 단순화시킬 수 없음

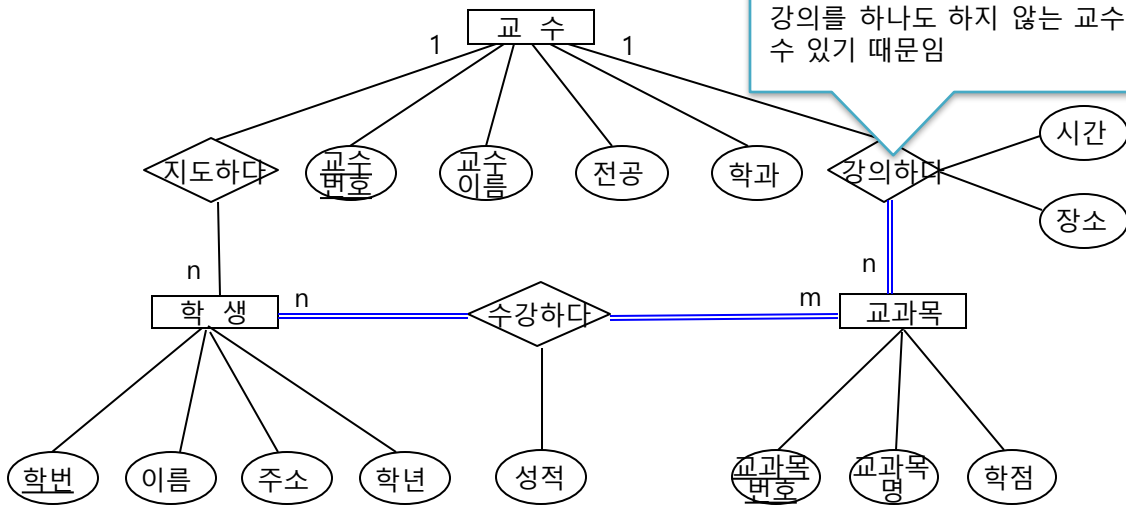
최종 릴레이션 스키마

상품(상품번호, 상품명, 상하의구분, 사이즈)
고객(id, pw, 주민등록번호, 이름, 휴대폰, 주소, 배송지, 구매횟수)
거래처(거래처번호, 상호, 연락처, 주소, 대표자명, 주요취급품목)
판매하다(상품번호(FK), 고객id(FK), 번호, 판매가격, 수량, 판매일자, 배송일자)
구매하다(상품번호(FK), 거래처번호(FK), 구매가격, 수량, 구매일자)

2. 릴레이션 생성 사례

2) 학사 관리 DB의 릴레이션 스키마 작성

학사 관리 DB 구축을 위한 ERD



2. 릴레이션 생성 사례

2) 학사 관리 DB의 릴레이션 스키마 작성

학사 관리 DB 구축을 위한 ERD

개체 릴레이션 생성 및
기본 키 설정

교수(교수번호, 교수이름, 전공, 학과)
학생(학번, 이름, 주소, 학년)
교과목(교과목번호, 교과목명, 학점)

관계 릴레이션 생성 및
기본 키 설정

지도하다(학번(FK), 교수번호(FK))
강의하다(교과목번호(FK), 교수번호(FK), 시간, 장소)
수강하다(학번(FK), 교과목번호(FK), 성적)

릴레이션 스키마
단순화

- a. '지도하다' 관계 릴레이션의 단순화
1 : N 유형이므로 N측 개체인 학생 개체 릴레이션에 1측 개체인 교수 개체 릴레이션의 기본 키인 '교수번호'를 외래 키로 포함시키고, '지도하다' 관계 릴레이션은 삭제한다.

학생(학번, 이름, 주소, 학년, 교수번호(FK))

- b. '강의하다' 관계 릴레이션의 단순화
1 : N 유형이므로 N측 개체인 교과목 개체 릴레이션에 1측 개체인 교수 개체 릴레이션의 기본 키인 '교수번호'를 외래 키로 포함시키고, '강의하다' 관계 릴레이션은 삭제한다.

교과목(교과목번호, 교과목명, 학점, 교수번호(FK))

- c. '수강하다' 관계 릴레이션의 단순화
M : N 유형이므로 단순화시킬 수 없다.

최종 릴레이션 스키마

교수(교수번호, 교수이름, 전공, 학과)
학생(학번, 이름, 주소, 학년, 교수번호(FK))
교과목(교과목번호, 교과목명, 학점, 교수번호(FK))
수강하다(학번(FK), 교과목번호(FK), 성적)

릴레이션 변환 초기에는 여섯 개의 릴레이션이 생성되었는데,
단순화 과정을 통해서 네 개의 릴레이션으로 줄어든 것을 알 수 있음

3. 무결성 제약조건 정의

1) 무결성 제약조건(Integrity Constraints)

무결성 제약조건(Integrity Constraints)이란?

DB에 저장된 데이터의 **정확성과 일관성** 유지를 위해서 DB가 항상 만족해야 하는 조건을 의미

관계 데이터 모델의 무결성 제약조건

구분	내용
개체 무결성 (Entity Integrity)	기본 키는 튜플들을 유일하게 식별하기 위해서 널 값을 가질 수 없다는 제약 조건
참조 무결성 (Referential Integrity)	외래 키는 반드시 피참조 릴레이션의 기본 키 값이나 널 값을 가져야 한다는 제약조건
의미 무결성 (Semantic Integrity)	특정 속성이 반드시 미리 정해진 값을 가져야 한다는 제약조건

2) 무결성 제약조건 정의 방법 구분 및 정의 기준

무결성 제약조건 정의 방법 구분

구분	내용
묵시적 정의	릴레이션을 정의할 때 기본 키와 외래 키를 정의하면, 개체 무결성 과 참조 무결성 이 묵시적으로 정의됨
명시적 정의	개체 무결성과 참조 무결성 이외에, 데이터의 정확성과 일관성 유지를 위해서 특정 속성이 만족해야 하는 무결성 제약조건(NOT NULL, UNIQUE, CHECK 등)은 명시적으로 정의해야 함

무결성 제약조건 정의 기준

1

ERD를 릴레이션 스키마로 변환하면서 **기본 키**와 **외래 키**가 식별되면 개체 무결성과 참조 무결성은 묵시적으로 정의되므로, 논리적 모델링 단계에서 별도로 정의할 필요는 없음

2

널 값 허용 여부(**NOT NULL 제약조건**), 고유 값 여부(**UNIQUE 제약조건**), 속성 값이 만족해야 하는 특정 조건(**CHECK 제약조건**) 등의 제약조건은 논리적 모델링 단계에서 명시적으로 정의함

3. 무결성 제약조건 정의

3) 무결성 제약조건 정의 사례

온라인 쇼핑몰 DB의 무결성 제약조건 정의 사례

No.	제약조건	제약 유형	관련 릴레이션
1	상품명과 사이즈는 널 값을 가질 수 없다.	NOT NULL	상품
2	상품명은 유일한 값을 가져야 한다.	UNIQUE	상품
3	pw와 주민등록번호, 이름, 휴대폰, 주소는 널 값을 가질 수 없다.	NOT NULL	고객
4	주민등록번호와 휴대폰은 유일한 값을 가져야 한다.	UNIQUE	고객
5	상호와 연락처는 널 값을 가질 수 없다.	NOT NULL	거래처
6	상호는 유일한 값을 가져야 한다.	UNIQUE	거래처
7	번호, 수량, 판매일자는 널 값을 가질 수 없다.	NOT NULL	판매한다
8	번호는 유일한 값을 가져야 한다.	UNIQUE	판매한다
9	수량, 구매일자는 널 값을 가질 수 없다.	NOT NULL	구매한다

학사 관리 DB의 무결성 제약조건 정의 사례

No.	제약조건	제약 유형	관련 릴레이션
1	교수이름, 전공, 학과는 널 값을 가질 수 없다.	NOT NULL	교수
2	이름과 학년은 널 값을 가질 수 없다.	NOT NULL	학생
3	주민등록번호와 휴대폰은 유일한 값을 가져야 한다.	UNIQUE	학생
4	교과목명, 학점, 교수번호는 널 값을 가질 수 없다.	NOT NULL	교과목
5	교과목명은 유일한 값을 가져야 한다.	UNIQUE	교과목

쉼터

꿈은 살아있는 사람의 의무이자 권리이다!

아들아! 죽는 날까지 꿈꾸기를 포기하지 마라.
매일 꿈을 꾸어라.
꿈꾸지 않는 사람은 아무것도 얻을 수 없으며,
오직 꿈꾸는 자만이 비상할 수 있다.
꿈에는 한계가 없다. 마음껏 꿈꿔라.
꿈을 꾸다는 것은 살아 있다는 증거이고
사람이 살아 있는 동안에 반드시 해야 할 의무이자 권리이다.

- 송길원, '나를 딛고 세상을 향해 뛰어올라라' 중에서

꿈을 꾸면 목표가 생기고,
목표를 잘게 나누면 계획이 되고,
계획을 하나씩 실행하면 꿈은 이루어집니다.

- 토지의 작가, 박경리