

C H A P T E R

08

데이터베이스 설계

Database Design

요구사항 분석부터 물리적 설계까지 — DB를 만드는 5단계 여정

박상돈 조교수

대전대학교 · 컴퓨터공학과 · AxGS Lab

CONTENTS

목차

01

데이터베이스 설계 단계

5단계 설계 프로세스 개요

02

요구 사항 분석

한빛 마트 사례 요구사항 명세서

03

개념적 설계

E-R 모델로 개체·관계 도출

04

논리적 설계

변환 규칙 5가지 + 한빛마트 종합

05

물리적 설계와 구현

테이블 명세서 → DDL

SECTION

01

데이터베이스 설계 단계

Database Design Process

DB를 어떻게 만들까? — 큰 그림 먼저

데이터베이스 설계란?

“ Definition

사용자들의 **요구 사항**을 고려하여 **데이터베이스를 생성**하는 과정

품질 좋은 데이터베이스의 3가지 조건



일관성·무결성

데이터가 항상 정확하고
모순 없이 유지된다



접근 편의성

사용자가 이해하기 쉽고
다루기 편해야 한다



요구사항 충족

구성원들이 원하는
기능을 만족해야 한다

데이터베이스 설계의 2가지 방법

METHOD 1

E-R 모델 + 변환 규칙



E-R Model & Relation Rules

- ▶ 요구사항을 **E-R 다이어그램**으로 그림
- ▶ **변환 규칙**을 적용해 릴레이션으로 변환
- ▶ **이번 장의 주된 접근법**

VS

METHOD 2

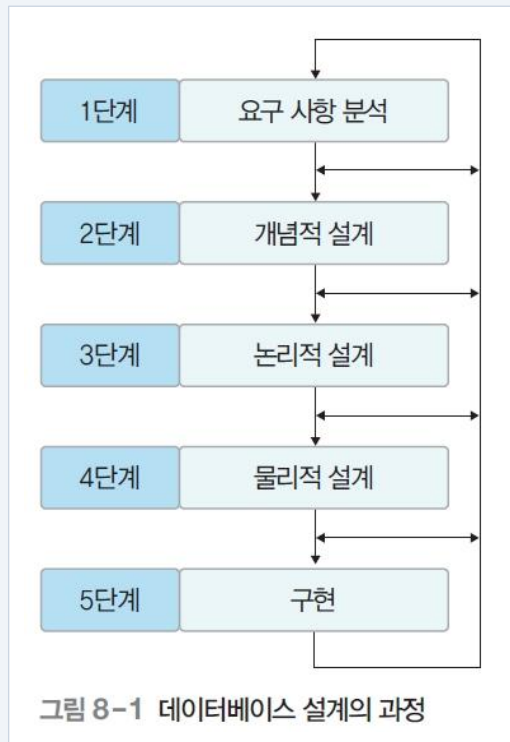
정규화 기법



Normalization

- ▶ 함수 종속성을 분석해 **릴레이션 분해**
- ▶ 데이터 중복과 **이상 현상 제거**
- ▶ **9장**에서 본격적으로 학습

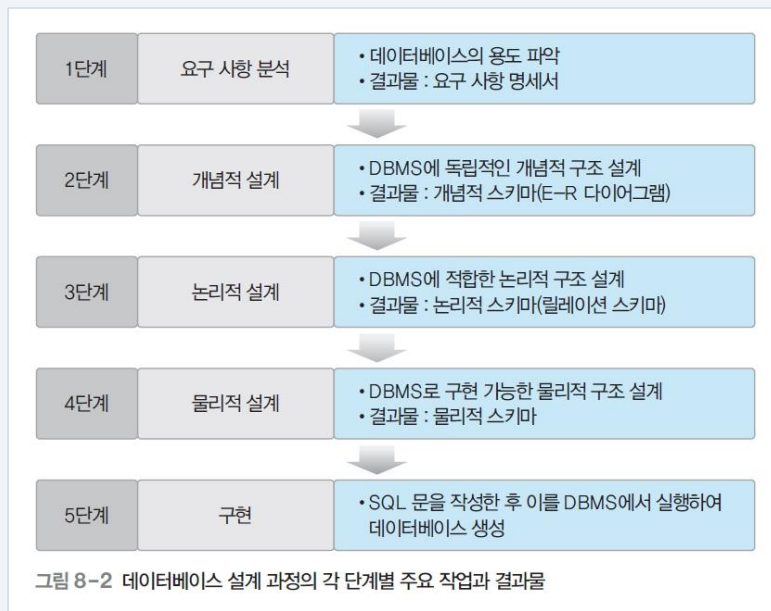
E-R 모델 기반 설계 5단계 — 전체 흐름



1	요구사항 분석	사용자 요구를 명세서로 작성
2	개념적 설계	DBMS 독립적 개념 스키마 (E-R)
3	논리적 설계	릴레이션 스키마로 변환
4	물리적 설계	저장 구조·인덱스 설계
5	구현	DDL로 DB 실제 생성

각 단계별 주요 작업과 결과물

단계마다 명확한 결과물(산출물)이 있고, 다음 단계의 입력이 됩니다



DB 설계의 핵심 3단계

5단계 중에서도 **가장 중요한 3단계**에 집중합니다

02*Requirements*
요구사항 분석

INPUT / 활동

사용자 요구사항 수집

OUTPUT / 산출물

요구 사항 명세서

03*Conceptual*
개념적 설계

INPUT / 활동

개체·속성·관계 추출

OUTPUT / 산출물

E-R 다이어그램

04*Logical*
논리적 설계

INPUT / 활동

변환 규칙 5가지 적용

OUTPUT / 산출물

릴레이션 스키마

SECTION

02

요구 사항 분석

Requirements Analysis

한빛 마트 사례 — 14개 요구사항 명세서

한빛 마트의 요구 사항 명세서

요구 사항 분석 · DB에 대한 사용자들의 요구사항을 수집·분석하여 명세서로 작성

- ❶ 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ❷ 가입한 회원에게는 등급과 적립금이 부여된다.
- ❸ 회원은 회원아이디로 식별한다.
- ❹ 상품에 대한 상품번호, 상품명, 재고량, 단가 정보를 유지해야 한다.
- ❺ 상품은 상품번호로 식별한다.
- ❻ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ❼ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.
- ❽ 각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ❾ 제조업체가 상품을 공급하면 공급일자와 공급량 정보를 유지해야 한다.
- ❿ 제조업체에 대한 제조업체명, 전화번호, 위치, 담당자 정보를 유지해야 한다.
- ⓫ 제조업체는 제조업체명으로 식별한다.
- ⓬ 회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.
- ⓭ 게시글에 대한 글번호, 글제목, 글내용, 작성일자 정보를 유지해야 한다.
- ⓮ 게시글은 글번호로 식별한다.

그림 8-4 한빛 마트의 데이터베이스를 위한 요구 사항 명세서

SECTION

03

개념적 설계

Conceptual Design

명세서를 E-R 다이어그램으로 — 명사·동사를 찾아내자

개념적 설계의 3단계 모델링 과정

요구 사항으로부터 **개체와 개체 간의 관계**를 분석하여 **E-R 다이어그램**으로 작성



개체·속성·키 선별

요구사항에서 명사를 찾아
개체와 속성으로 분류,
각 개체의 키 속성 지정



관계 결정

요구사항에서 동사를 찾아
개체 간의 관계 추출,
카디널리티와 참여 결정



E-R 다이어그램 작성

개체·속성·관계를
표준 기호로 시각화하여
개념적 스키마 완성

STEP 1 — 개체와 속성 추출: 명사를 찾아라



핵심 원칙

개체는 대체적으로 요구사항 내에서 **명사**를 찾아야 한다

01

명사가 여러 개면 **대표 명사 하나만** 선택



02

나머지 명사는 그 **개체의 속성**이 될 수 있음



03

조직 자체를 가리키는 명사 (**예: 한빛 마트**)는 제외



04

명사처럼 보여도 **실제 명사가 아닐 수 있음** (예: 동명사)



사례 분석 ① — 회원 명사 선별 → 개체·속성 분류

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ② 가입한 회원에게는 등급과 적립금이 부여된다.
- ③ 회원은 회원아이디로 식별한다.

그림 8-7 요구 사항 문장에서 명사를 선별한 예 (A)

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ② 가입한 회원에게는 등급과 적립금이 부여된다.
- ③ 회원은 회원아이디로 식별한다.

그림 8-8 요구 사항 문장에서 개체와 속성을 추출한 예 (A)

사례 분석 ② — 주문 문장에서 개체와 속성 가려내기

⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.

그림 8-9 요구 사항 문장에서 명사를 선별한 예 (B)

⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.

그림 8-10 요구 사항 문장에서 개체와 속성을 추출한 예 (B)

 주의 주문번호·주문수량·배송지·주문일자는 회원에도, 상품에도 속할 수 없는 명사 → 나중에 '주문' 관계의 속성으로 처리

사례 분석 ③ — 한빛 마트 명세서 전체 명사 분류

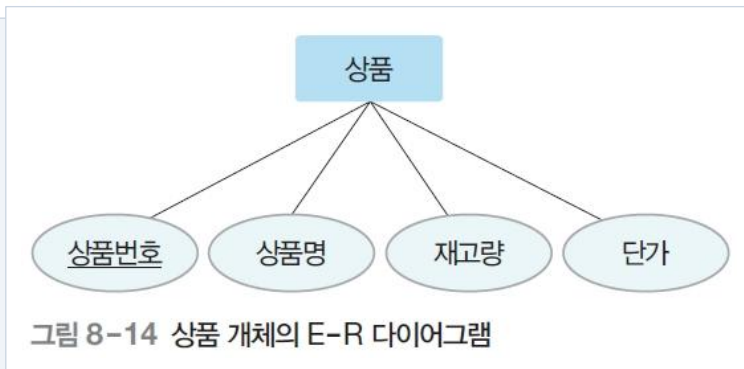
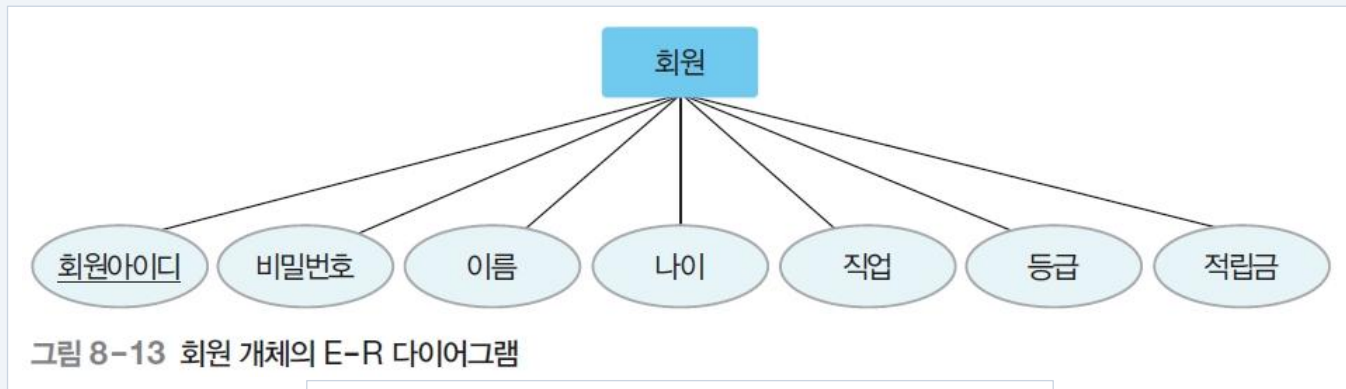
- ❶ 한빛 마트에 **회원**으로 가입하려면 **회원아이디**, **비밀번호**, **이름**, **나이**, **직업**을 입력해야 한다.
- ❷ 가입한 회원에게는 **등급**과 **적립금**이 부여된다.
- ❸ 회원은 회원아이디로 식별한다.
- ❹ **상품**에 대한 **상품번호**, **상품명**, **재고량**, **단가** 정보를 유지해야 한다.
- ❺ 상품은 상품번호로 식별한다.
- ❻ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ❼ 회원이 상품을 주문하면 주문에 대한 **주문번호**, **주문수량**, **배송지**, **주문일자** 정보를 유지해야 한다.
- ❽ 각 상품은 한 **제조업체**가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ❾ 제조업체가 상품을 공급하면 **공급일자**와 **공급량** 정보를 유지해야 한다.
- ❿ 제조업체에 대한 **제조업체명**, **전화번호**, **위치**, **담당자** 정보를 유지해야 한다.
- ⓫ 제조업체는 제조업체명으로 식별한다.
- ⓬ 회원은 **게시글**을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.
- ⓭ 게시글에 대한 **글번호**, **글제목**, **글내용**, **작성일자** 정보를 유지해야 한다.
- ⓮ 게시글은 글번호로 식별한다.

그림 8-11 한빛 마트 요구 사항 명세서에서 명사를 선별한 결과

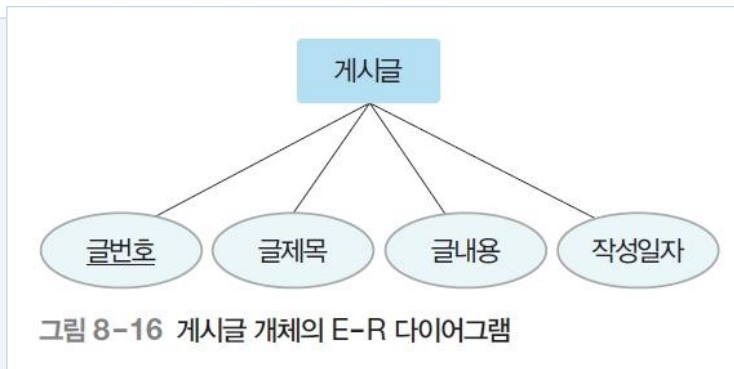
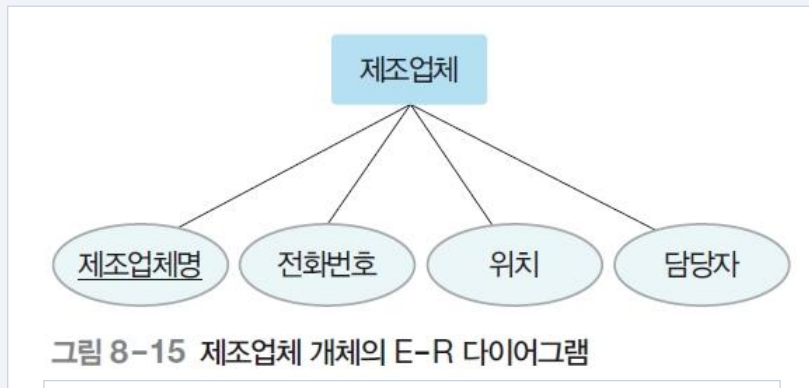
개체	속성
회원	회원아이디, 비밀번호, 이름, 나이, 직업, 등급, 적립금
상품	상품번호, 상품명, 재고량, 단가
제조업체	제조업체명, 전화번호, 위치, 담당자
게시글	글번호, 글제목, 글내용, 작성일자

그림 8-12 한빛 마트 요구 사항 명세서에서 개체와 개체의 속성을 추출한 최종 결과

각 개체별 E-R 다이어그램 — 회원 / 상품



각 개체별 E-R 다이어그램 — 제조업체 / 게시물



STEP 2 — 관계 추출: 동사를 찾아라



핵심 원칙

관계는 대체적으로 요구사항 내에서 **동사**를 찾아야 한다

예시 — 요구사항 1~3번 동사 추출

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 **입력해야 한다**.
- ② 가입한 회원에게는 등급과 적립금이 **부여된다**.
- ③ 회원은 회원아이디로 **식별한다**.

그림 8-17 요구 사항 문장에서 동사를 선별한 예 (A)

- ▶ 동사가 여러 개면 **대표 동사 하나만** 선택
- ▶ **매핑 카디널리티**: 1:1 / 1:n / n:m 결정
- ▶ **참여 특성**: 필수(이중 실선) / 선택(단일 실선) 결정
- ▶ **주의**: '입력해야 한다', '부여된다', '식별한다', '유지해야 한다'는 단순 설명 → 관계 아님

매핑 카디널리티(Mapping Cardinality)

 $1 : 1$

일대일

One-to-One

한 개체가 다른 한 개체와만

EXAMPLES

남편 ↔ 아내
사람 ↔ 주민등록번호

 $1 : n$

일대다

One-to-Many

한 쪽이 여러 쪽을 가짐

EXAMPLES

제조업체 ↔ 상품
회원 ↔ 게시글

 $n : m$

다대다

Many-to-Many

양쪽 모두 여러 개와 연결

EXAMPLES

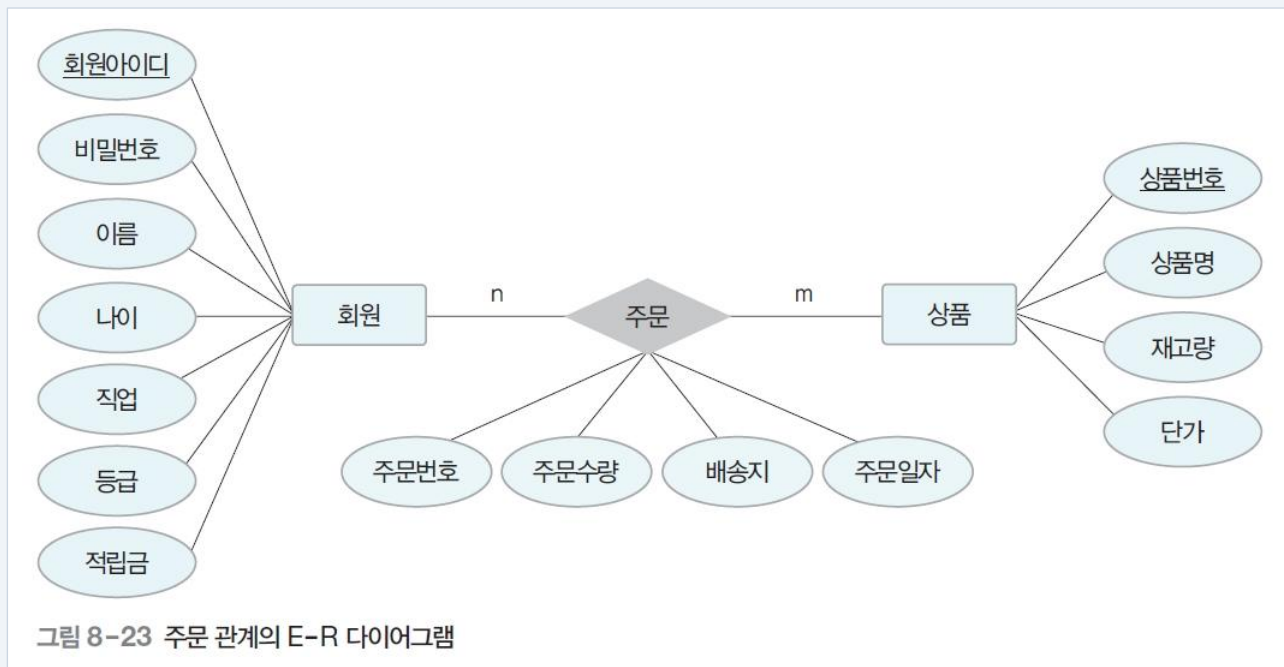
학생 ↔ 과목
회원 ↔ 상품(주문)



참여 특성 → 필수 참여 (이중 실선) / 선택 참여 (단일 실선)

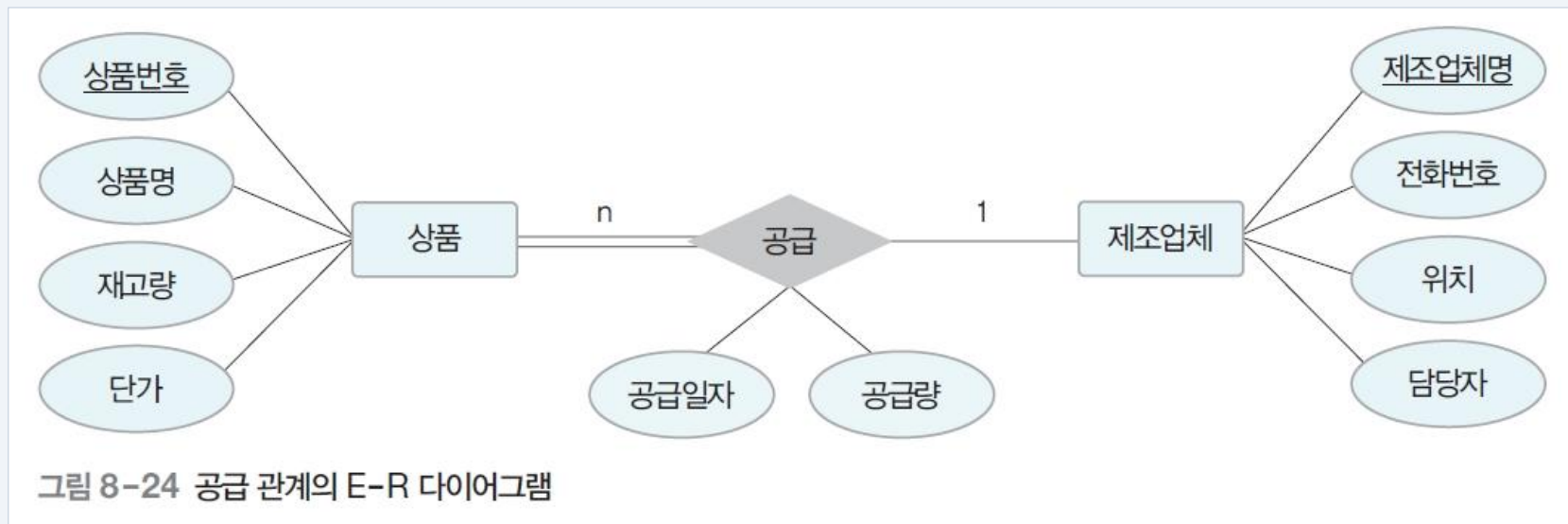
한빛 마트 관계 ① — 주문 (회원 n : 상품 m)

요구사항 6번 "회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다" → n:m



한빛 마트 관계 ② – 공급 (1:n, 상품 필수 참여)

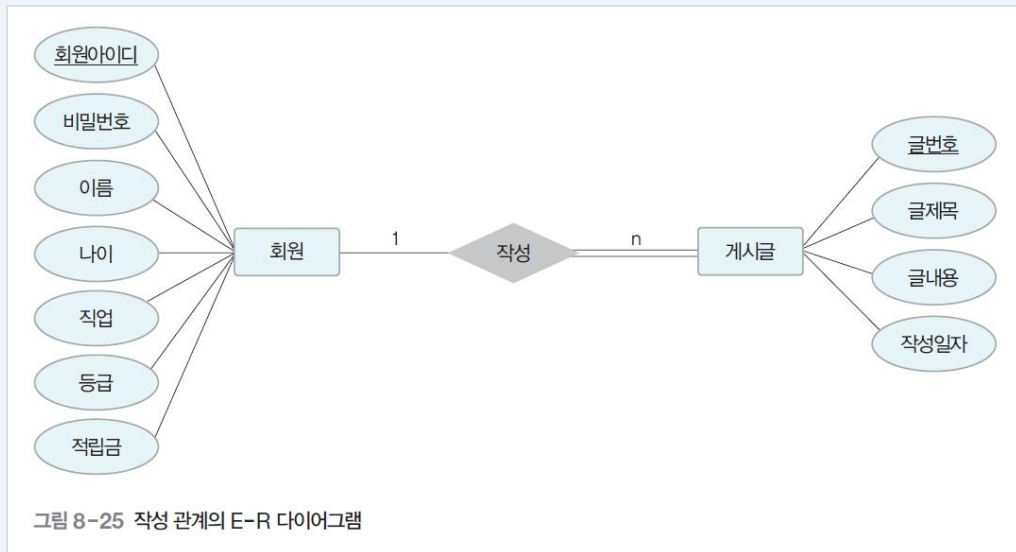
요구사항 8번 "각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다" → 1:n + 상품은 필수 참여



i 이중 실선 상품 쪽이 이중 실선 → "모든 상품은 반드시 어떤 제조업체에서 공급된 것이어야 한다"

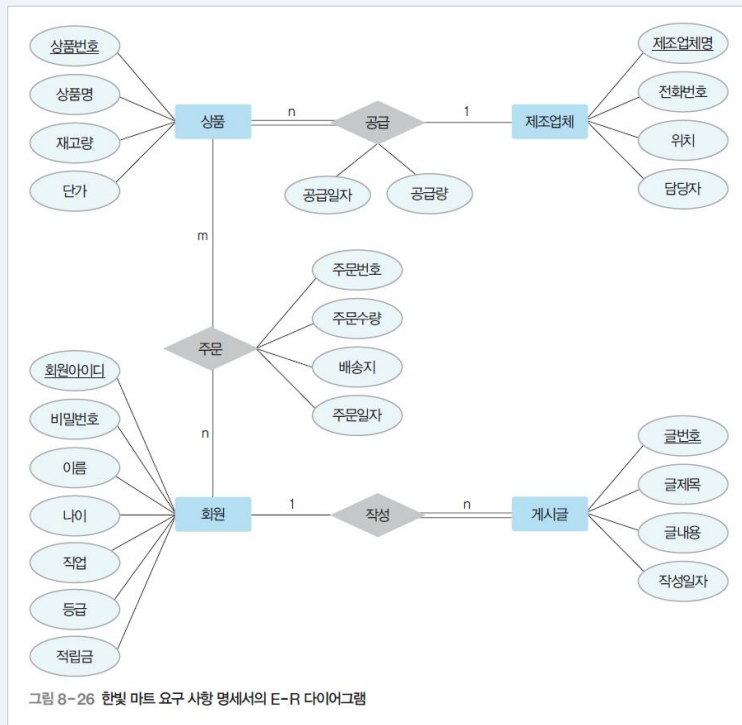
한빛 마트 관계 ③ — 작성 (1:n, 게시물 필수 참여)

요구사항 12번 "회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다" → 1:n



관계 속성 없음 작성일자는 이미 게시글 개체의 속성이므로, 작성 관계는 별도 속성을 갖지 않음

STEP 3 — 한빛 마트 E-R 다이어그램 완성 ★



SECTION

04

논리적 설계

Logical Design

E-R 다이어그램 → 릴레이션 스키마 — 5가지 변환 규칙

E-R 다이어그램 → 릴레이션, 무엇이 다른가?

구분	E-R 다이어그램	관계 데이터 모델
개체와 관계 구분 여부	구분함 (사각형 / 마름모)	구분하지 않음 모두 릴레이션으로 표현
다중 값 속성 표현 허용	허용 (이중 타원)	허용하지 않음 (별도 릴레이션 필요)
복합 속성 표현 허용	허용 (속성에 하위 속성)	허용하지 않음 (단순 속성으로 분해)



결론 차이가 있는 만큼 **변환 규칙**이 필요하다 → 5가지 규칙을 학습

릴레이션 스키마 변환 규칙 5가지

01



모든 개체

→ 릴레이션으로 변환

02



n:m 관계

→ 별도의 릴레이션으로 변환

03



1:n 관계

→ 외래키(Foreign Key)로 표현

04



1:1 관계

→ 외래키로 표현 (서로 / 단방향 / 통합)

05



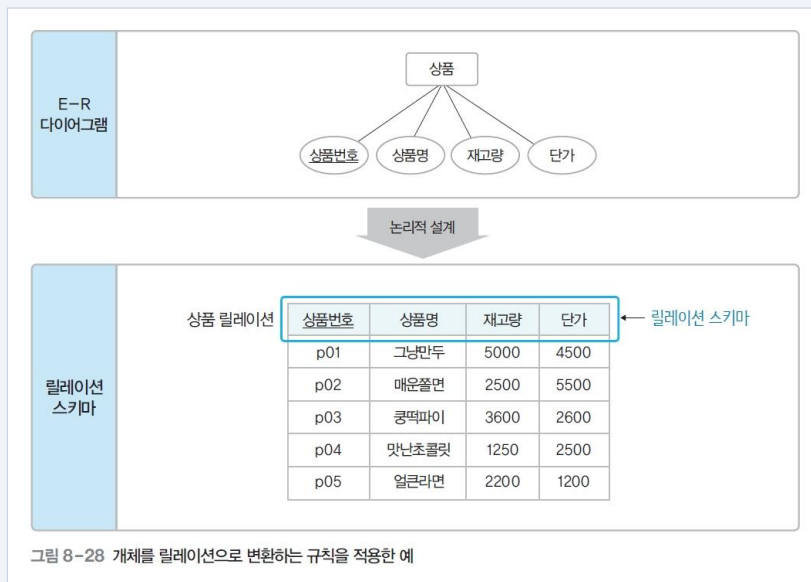
다중 값 속성

→ 별도의 릴레이션으로 변환

◇ + 기타 고려 사항: 속성이 많은 관계, 순환(자기 참조) 관계 등 예외 케이스

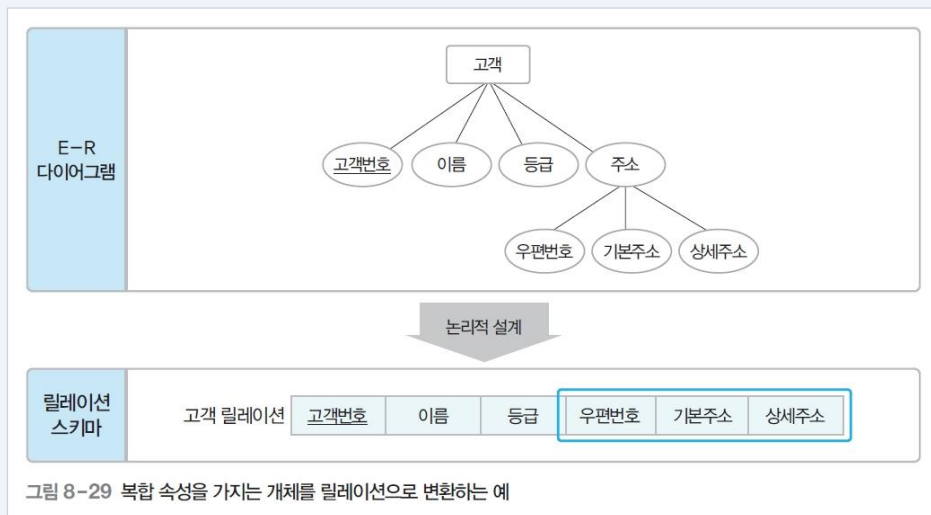
규칙 1 — 모든 개체는 릴레이션으로

RULE 01 | 모든 개체는 릴레이션으로 변환한다



규칙 1 (응용) — 복합 속성은 단순 속성으로 분해

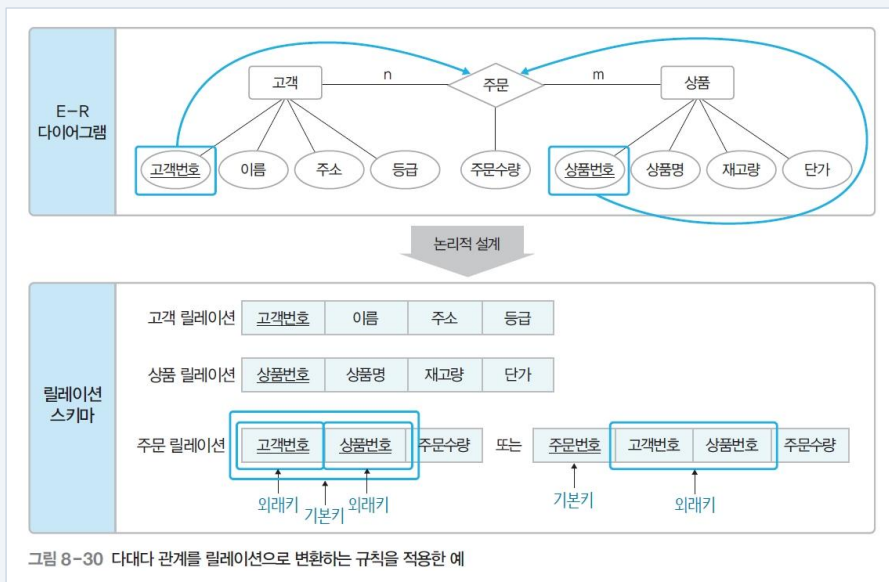
RULE 01-b | 복합 속성은 단순 속성으로 분해하여 릴레이션에 포함



주소(복합) → 우편번호 · 기본주소 · 상세주소 **3개 단순 속성**으로 분해 → 검색·정렬이 쉬워짐

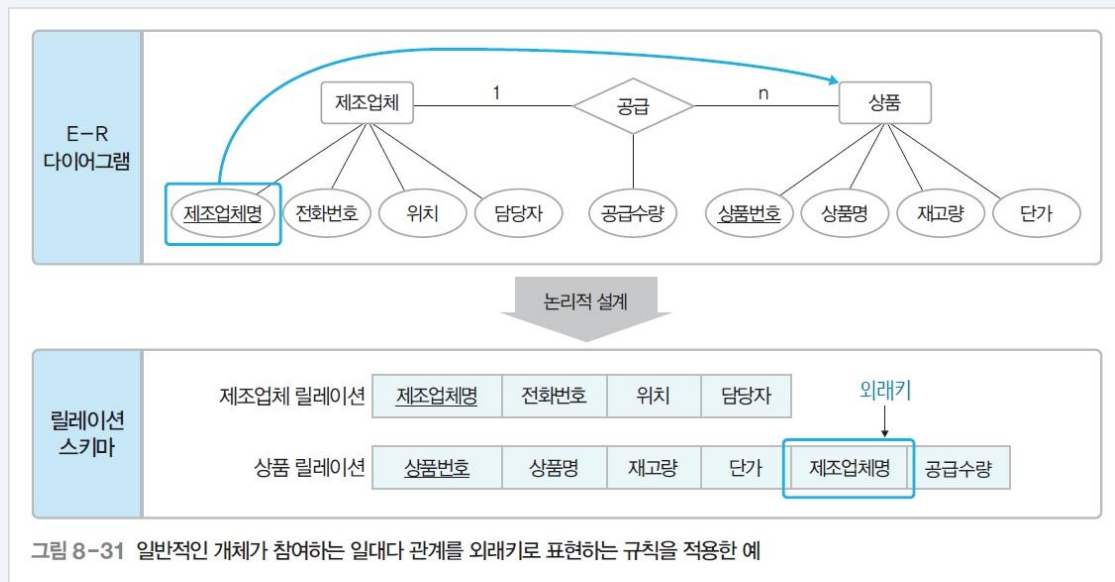
규칙 2 — n:m 관계는 별도 릴레이션으로

RULE 02 | 다대다 관계는 별도의 릴레이션으로 변환한다



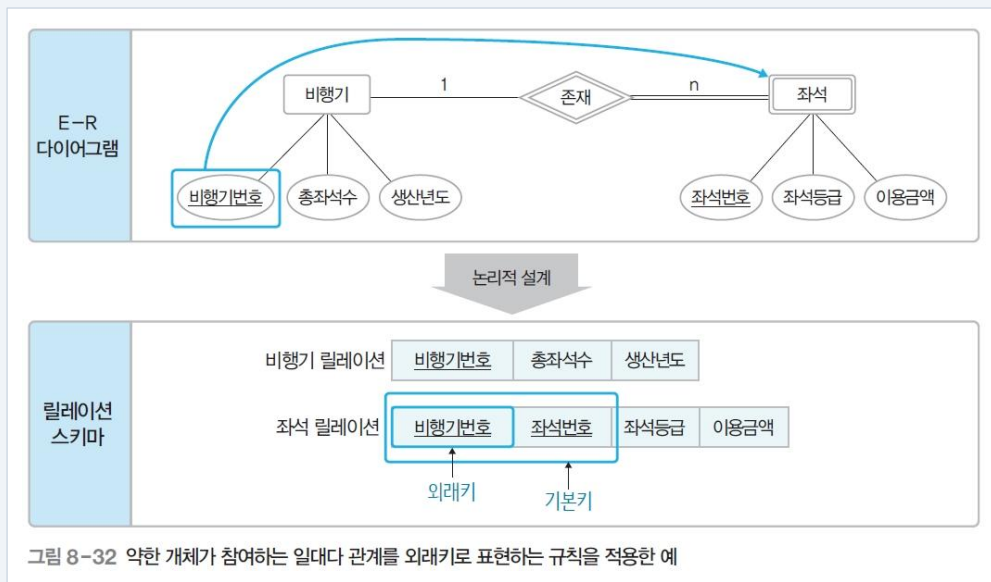
규칙 3-1 — 일반적인 1:n 관계는 외래키로

RULE 03-1 | 1:n 관계 → n 쪽 릴레이션에 1 쪽의 기본키를 외래키로 추가



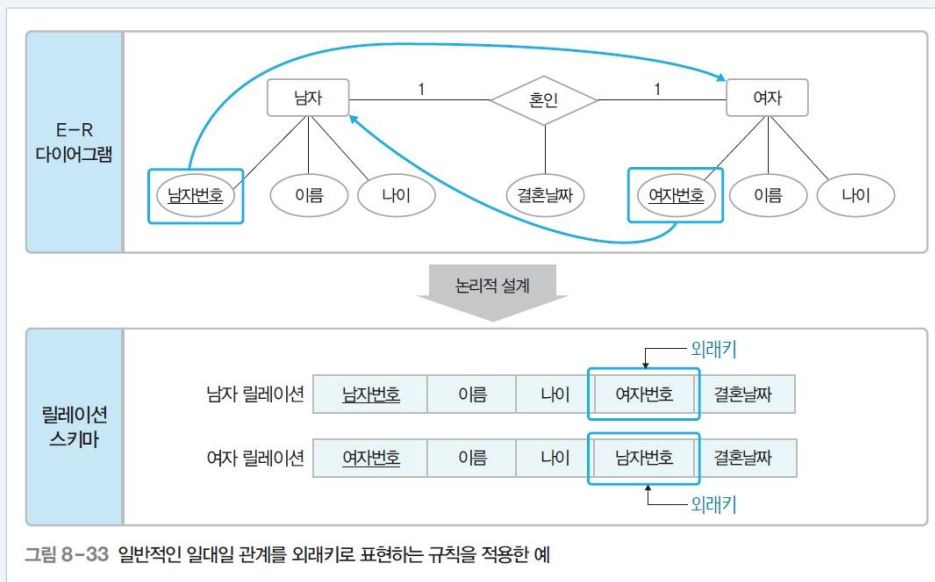
규칙 3-2 — 약한 개체의 1:n 관계

RULE 03-2 | 약한 개체 → 외래키를 **복합 기본키**에 포함



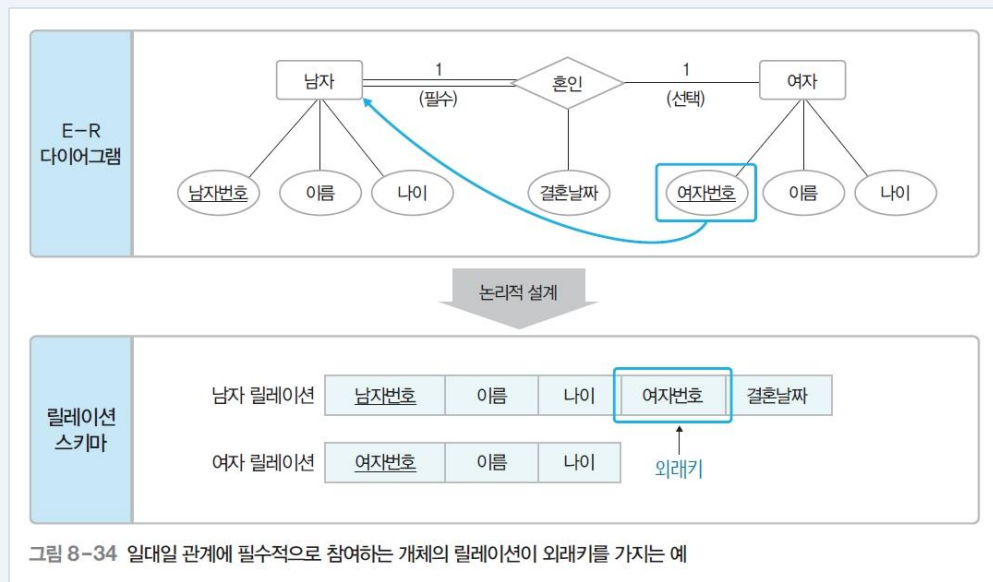
규칙 4-1 — 일반적인 1:1 관계 (양쪽 선택)

RULE 04-1 | 양쪽 모두 선택 참여 → 서로의 PK를 외래키로 주고받음



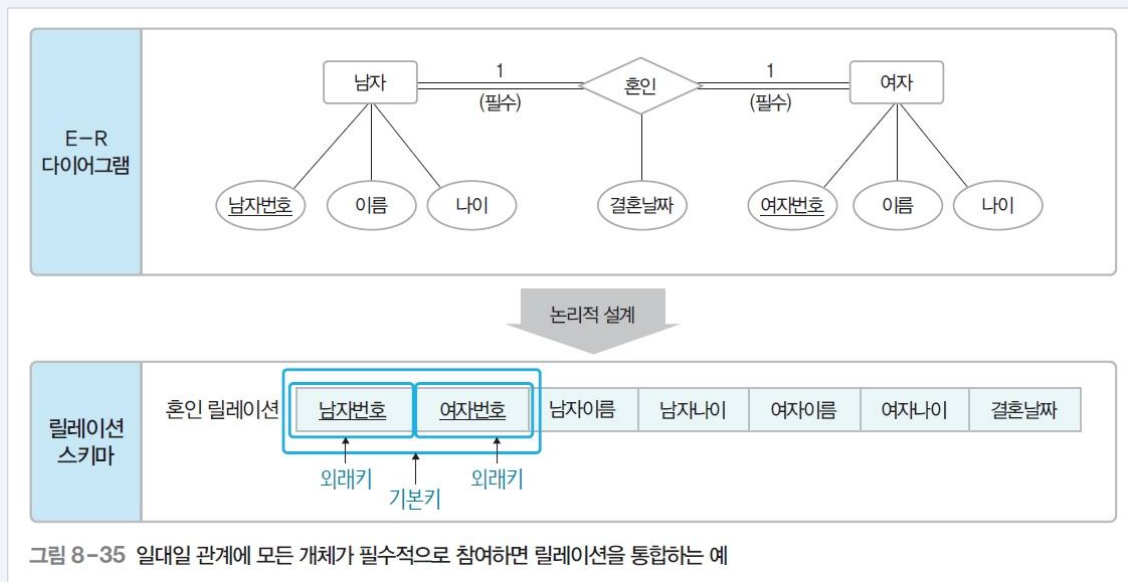
규칙 4-2 — 한쪽만 필수 참여하는 1:1

RULE 04-2 | 한쪽만 필수 참여 → 필수 참여하는 쪽만 외래키를 가짐



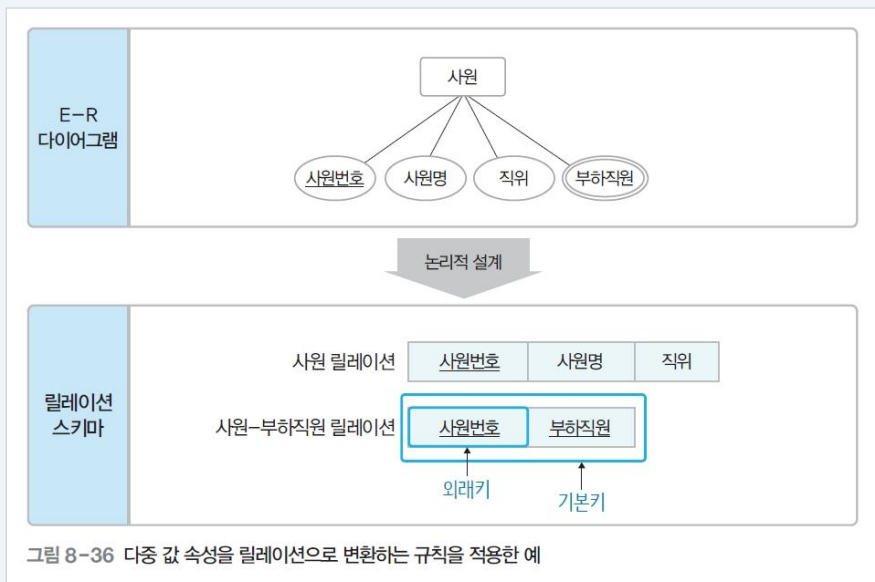
규칙 4-3 — 양쪽 모두 필수 참여 → 릴레이션 통합

RULE 04-3 | 양쪽 모두 필수 참여 → 두 릴레이션을 하나로 합침



규칙 5 — 다중 값 속성은 별도 릴레이션으로

RULE 05 | 다중 값 속성(이중 타원)은 별도의 릴레이션으로 변환한다



규칙 5 (상세) — 다중 값을 처리하는 두 가지 방법

BAD — 한 셀에 다중 값

사원번호	사원명	직위	부하직원
e001	홍정화	부장	{김정수, 이수연}
e002	김수창	과장	{박영길}
e003	최종민	차장	{이수영, 배길수}

← 다중 값을 가지는 속성

그림 8-37 다중 값 속성인 부하직원 속성을 그대로 포함하는 사원 릴레이션

GOOD — 별도 릴레이션 분리

사원 릴레이션			사원-부하직원 릴레이션	
사원번호	사원명	직위	사원번호	부하직원
e001	홍정화	부장	e001	김정수
e002	김수창	과장	e001	이수연
e003	최종민	차장	e002	박영길
			e003	이수영
			e003	배길수

그림 8-39 규칙 5를 적용한 후의 사원 릴레이션과 사원-부하직원 릴레이션

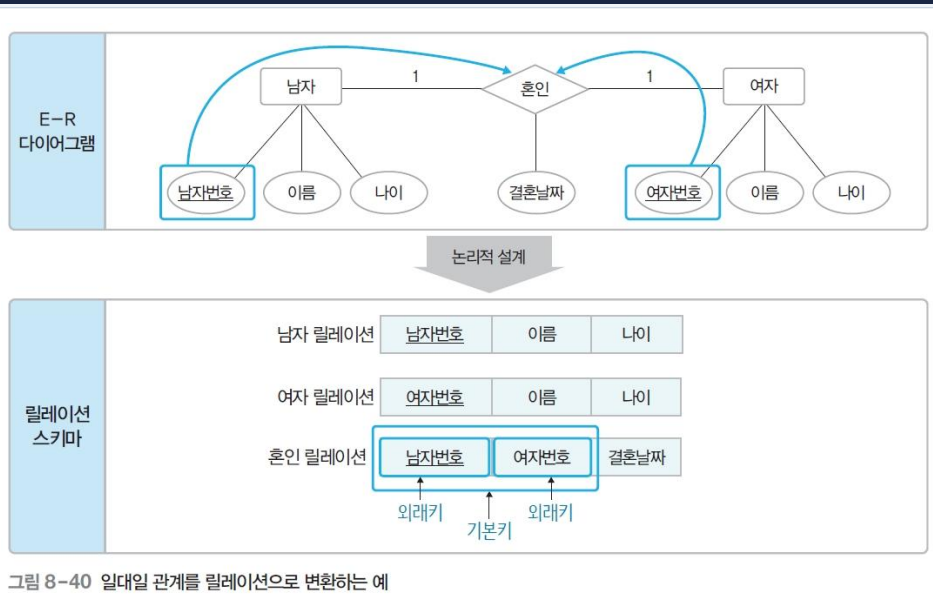


왜 별도 릴레이션으로 분리해야 하는가?

- ▶ **1NF(제1정규형)** 준수: 각 셀은 단일 값
- ▶ **검색·갱신 효율**: LIKE '%값%' 같은 비효율 회피, 인덱스 활용 가능
- ▶ **무결성·확장성**: INSERT/DELETE 깔끔, 추가 속성도 쉽게 부여 가능

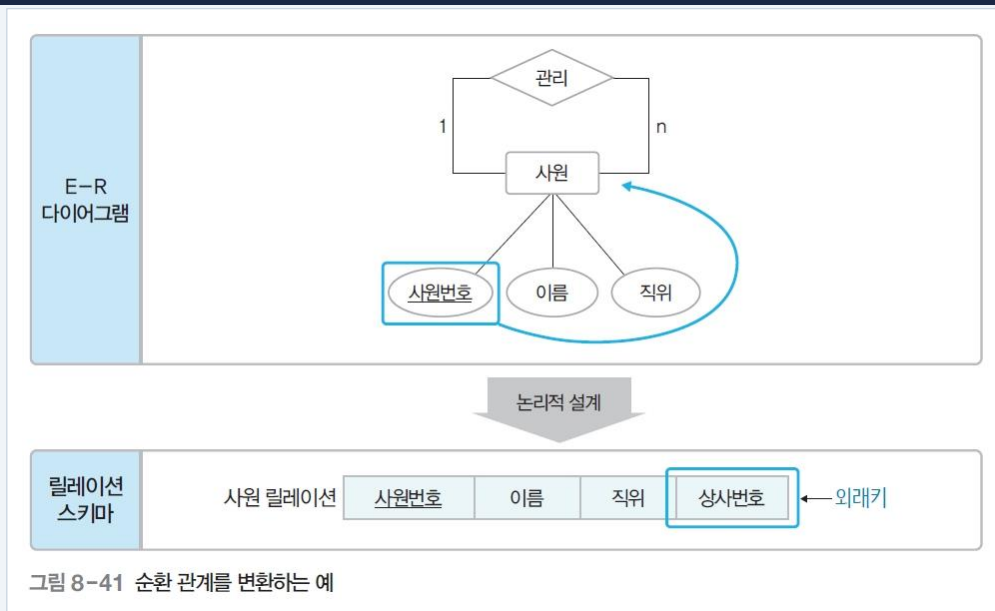
기타 고려 ① — 속성이 많은 관계의 변환

원칙 속성이 많은 관계는 관계 유형(1:n, 1:1)과 **상관없이 릴레이션으로 변환** 가능 — 단, n:m이 아닌데 별도 릴레이션을 만들면 JOIN 증가로 DBMS 부담



기타 고려 ② — 순환(자기 참조) 관계

i 원칙 자기 자신을 참조하는 **1:n 순환 관계** → 같은 릴레이션 내에 외래키로 포함 (별도 릴레이션 불필요)



한빛 마트 — E-R → 릴레이션 스키마 변환 결과 ★



■ 5개 릴레이션 | 회원 · 상품(+제조업체명 FK) · 제조업체 · 게시글(+회원아이디 FK) · 주문(복합 PK)

테이블 명세서 작성하기 — 회원 릴레이션 예

테이블 명세서 · 릴레이션 스키마에 대한 **설계 정보를 기술한 문서** (MS SQL Server 기준)

테이블 이름			회원			
속성 이름	데이터 타입	널 허용 여부	기본값	기본키	외래키	제약조건
회원아이디	VARCHAR(20)	N		PK		
비밀번호	VARCHAR(20)	N				
이름	VARCHAR(10)	N				
나이	INT	Y				0 이상
직업	VARCHAR(20)	Y				
등급	VARCHAR(10)	N	silver			silver, gold, vip만 허용
적립금	INT	N	0			

그림 8-48 회원 릴레이션 스키마의 테이블 명세서

SECTION

05

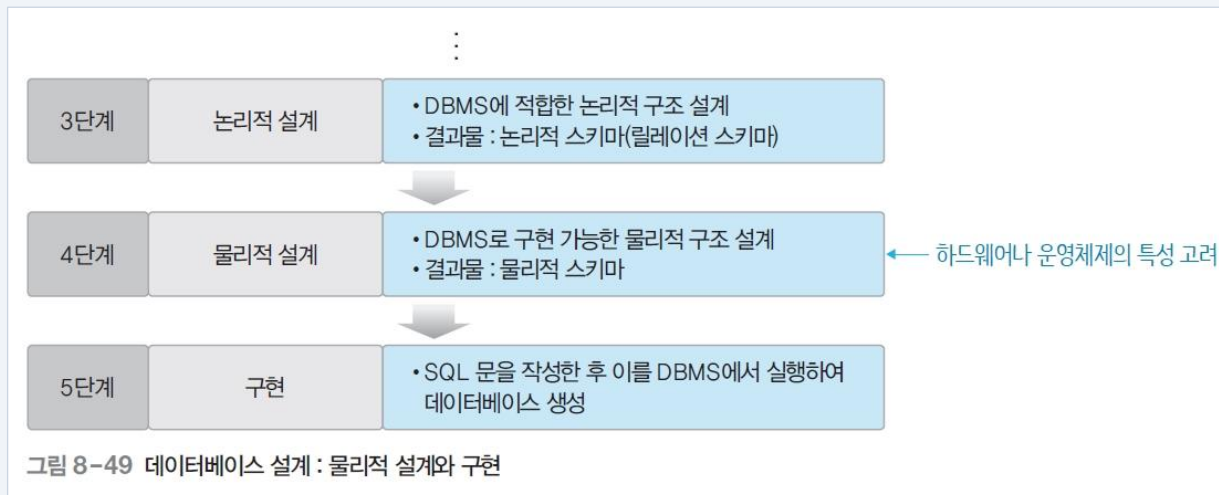
물리적 설계와 구현

Physical Design & Implementation

테이블 명세서를 실제 DB로 — DDL 작성

물리적 설계 → 구현으로의 흐름

물리적 설계 DBMS로 구현 가능한 물리적 구조 설계 (하드웨어/OS 특성 고려) → **구현** SQL 문 작성 후 DBMS에서 실행하여 DB 생성



현실 대부분의 물리적 결정은 DBMS가 자동 처리 — 입문 단계에서는 **PK 자동 인덱스 + 자주 검색하는 컬럼에 인덱스** 정도만 알면 알면 충분

구현 — DDL로 데이터베이스 빌드

DDL (Data Definition Language)로 테이블 명세서를 실제 DB 객체로 변환

```
CREATE TABLE 회원 (  
    회원아이디    VARCHAR(20)    NOT NULL,  
    비밀번호      VARCHAR(20)    NOT NULL,  
    이름          VARCHAR(10)    NOT NULL,  
    나이          INT,  
    직업          VARCHAR(20),  
    등급          VARCHAR(10)    NOT NULL DEFAULT 'silver',  
    적립금        INT            NOT NULL DEFAULT 0,  
    PRIMARY KEY(회원아이디),  
    CHECK (나이 >= 0),  
    CHECK (등급 in ('silver', 'gold', 'vip'))  
);
```

그림 8-50 회원 릴레이션을 구현하기 위한 SQL 문의 예



워크플로우 명세서 → SQL → 빌드 → 검증



필수 DB 빌드 스크립트는 반드시 백업

오늘 배운 내용 정리

01	 DB 설계 5단계	요구사항 → 개념적 → 논리적 → 물리적 → 구현
02	 개념적 설계 = 명사·동사 찾기	명사 → 개체·속성 / 동사 → 관계
03	 카디널리티 3가지	1:1 / 1:n / n:m + 참여 특성(필수/선택)
04	 변환 규칙 5가지	개체 → 릴레이션, n:m → 릴레이션, 1:n → FK, 1:1 → FK(3케이스), 다중값 → 릴레이션
05	 명세서 → DDL → 백업	테이블 명세서로 CREATE TABLE 작성 후 스크립트 반드시 백업



한 줄 요약

사람의 요구 → 그림(E-R) → 표(릴레이션) → 명세서 → SQL → 진짜 DB

T H A N K Y O U

수고하셨습니다

8장 데이터베이스 설계 · 끝

N E X T C L A S S

Chapter 09 · 정규화 (Normalization)

함수 종속성 · 이상 현상 · 1NF~BCNF · 정규화 vs 역정규화

8장에서 만든 릴레이션을 더 깔끔하게 다듬는 작업입니다

👉 다음 시간에 만나요!

박상돈 조교수 · 대전대학교 컴퓨터공학과 · AxGS Lab