



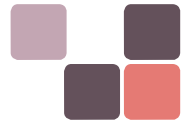
## ❖ 데이터베이스 설계

- 사용자의 다양한 요구 사항을 고려하여 데이터베이스를 생성하는 과정

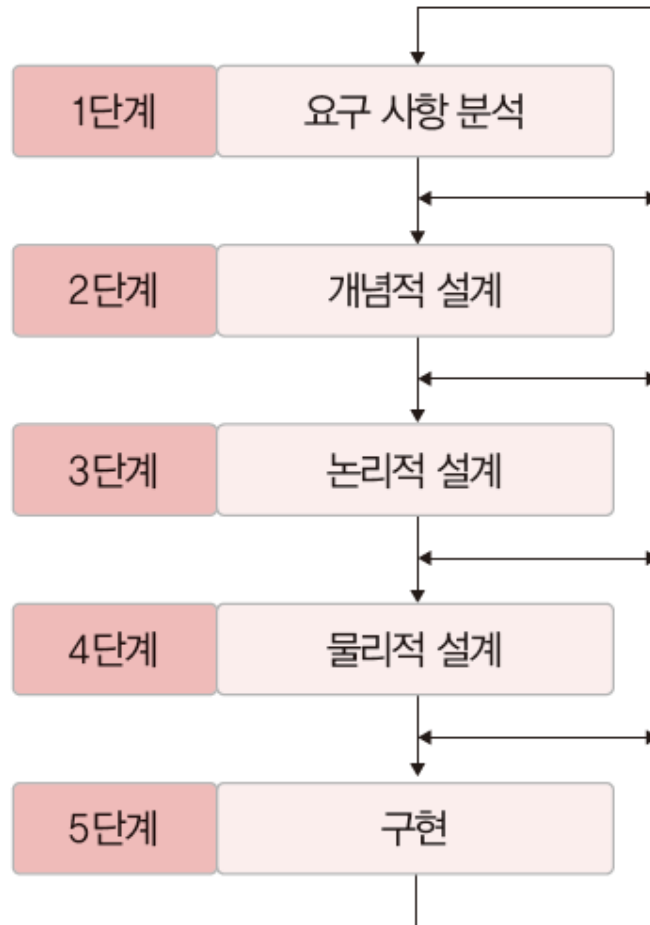
## ❖ 관계 데이터베이스의 대표적인 설계 방법

- E-R 모델과 릴레이션 변환 규칙을 이용한 설계
- 정규화를 이용한 설계

# 01 데이터베이스 설계 단계



## ❖ E-R 모델과 릴레이션 변환 규칙을 이용한 설계의 과정



설계 과정 중에 오류를 발견하여  
변경이 필요하다면 이전 단계로  
되돌아가 설계 내용을 변경 가능

그림 8-1 데이터베이스 설계의 과정

# 01 데이터베이스 설계 단계



## ❖ E-R 모델과 릴레이션 변환 규칙을 이용한 설계의 과정



그림 8-2 데이터베이스 설계 과정의 각 단계별 주요 작업과 결과물



### ❖ 설계 1 단계 : 요구 사항 분석

#### ■ 목적

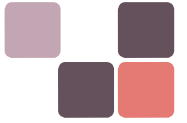
- 사용자의 요구 사항을 수집하고 분석하여 개발할 데이터베이스의 용도를 파악
  - 업무에 필요한 데이터가 무엇인지, 그 데이터에 어떤 처리가 필요한지 등을 고려

#### ■ 결과물

- 요구 사항 명세서

#### ■ 주요 작업

- 데이터베이스를 실제로 사용할 주요 사용자의 범위를 결정
- 사용자가 조직에서 수행하는 업무를 분석
- 면담, 설문 조사, 업무 관련 문서 분석 등의 방법을 이용해 요구 사항 수집
- 수집된 요구 사항에 대한 분석 결과를 요구 사항 명세서로 작성



### ❖ 요구 사항 분석 예 – [한빛 마트 데이터베이스]

#### ■ 인터넷으로 회원들에게 상품을 판매하는 한빛 마트의 데이터베이스 개발

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ② 가입한 회원에게는 등급과 적립금이 부여된다.
- ③ 회원은 회원아이디로 식별한다.
- ④ 상품에 대한 상품번호, 상품명, 재고량, 단가 정보를 유지해야 한다.
- ⑤ 상품은 상품번호로 식별한다.
- ⑥ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.
- ⑧ 각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ⑨ 제조업체가 상품을 공급하면 공급일자와 공급량 정보를 유지해야 한다.
- ⑩ 제조업체에 대한 제조업체명, 전화번호, 위치, 담당자 정보를 유지해야 한다.
- ⑪ 제조업체는 제조업체명으로 식별한다.
- ⑫ 회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.
- ⑬ 게시글에 대한 글번호, 글제목, 글내용, 작성일자 정보를 유지해야 한다.
- ⑭ 게시글은 글번호로 식별한다.

## 02 요구 사항 분석



그림 8-3 데이터베이스 설계 : 요구 사항 분석 단계



## ❖ 설계 2 단계 : 개념적 설계

### ■ 목적

- DBMS에 독립적인 개념적 스키마 설계
- 요구 사항 분석 결과물을 개념적 데이터 모델을 이용해 개념적 구조로 표현
  - ➔ 개념적 모델링
  - 일반적으로 E-R 모델을 많이 이용

### ■ 결과물

- 개념적 스키마 : E-R 다이어그램

### ■ 주요 작업

- 요구 사항 분석 결과를 기반으로 중요한 개체를 추출하고 개체 간의 관계를 결정하여 E-R 다이어그램으로 표현



### ❖ 설계 2 단계 : 개념적 설계

#### ■ 작업 과정

- STEP 1) 개체 추출, 각 개체의 주요 속성과 키 속성 선별
- STEP 2) 개체 간의 관계 결정
- STEP 3) E-R 다이어그램으로 표현

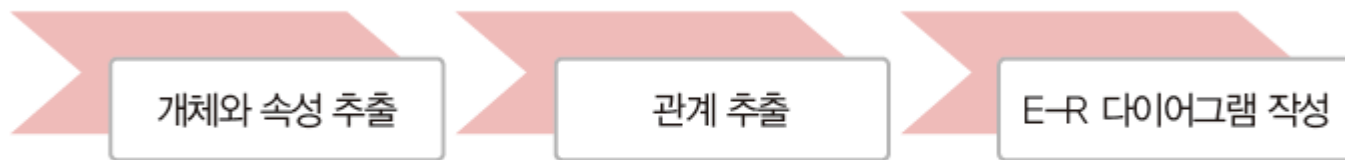


그림 8-6 개념적 모델링 과정





### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출

- 개체 : 저장할 만한 가치가 있는 중요 데이터를 가진 사람이나 사물 등
  - 예) 병원 데이터베이스 개발에 필요한 개체
    - 병원 운영에 필요한 사람 : 환자, 의사, 간호사 등
    - 병원 운영에 필요한 사물 : 병실, 수술실, 의료 장비 등
- 개체 추출 방법
  - 요구 사항 문장에서 업무와 관련이 깊은 의미 있는 **명사**를 찾아라!
    - 업무와 관련이 적은 일반적이고 광범위한 의미의 명사는 제외
    - 의미가 같은 명사가 여러 개일 경우는 대표 명사 하나만 선택
  - 찾아낸 명사를 개체와 속성으로 분류하라!



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출하는 과정

- 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)

- "한빛 마트"는 일반적이고 광범위한 의미의 명사이므로 제외
- "회원아이디", "비밀번호", "이름", "나이", "직업", "등급", "적립금"은 회원의 속성으로 분류
- "회원아이디"는 키 속성으로 분류

## 03 개념적 설계



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출하는 과정

- ① 한빛 마트에 **회원**으로 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)

#### [추출 결과]

- 개체 : 회원
- “회원” 개체의 속성 : 회원아이디, 비밀번호, 이름, 나이, 직업, 등급, 적립금
- “회원” 개체의 키 속성 : 회원아이디

## 03 개념적 설계



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

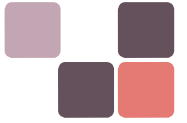
- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출하는 과정

⑦ **회원**이 **상품**을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.

그림 8-10 요구 사항 문장에서 개체와 속성을 추출한 예 (B)

#### [추출 결과]

- 개체 : 회원, 상품
- 속성 : 주문번호, 주문수량, 배송지, 주문일자
  - 회원이 상품을 주문을 해야 생기는 중요한 정보이기 때문에 회원이나 상품 개체의 속성으로 보기는 어렵고 이후 추출할 특정 관계의 속성일 가능성이 높음



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출하는 과정

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ② 가입한 회원에게는 등급과 적립금이 부여된다.
- ③ 회원은 회원아이디로 식별한다.
- ④ 상품에 대한 상품번호, 상품명, 재고량, 단가 정보를 유지해야 한다.
- ⑤ 상품은 상품번호로 식별한다.
- ⑥ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.
- ⑧ 각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ⑨ 제조업체가 상품을 공급하면 공급일자와 공급량 정보를 유지해야 한다.
- ⑩ 제조업체에 대한 제조업체명, 전화번호, 위치, 담당자 정보를 유지해야 한다.
- ⑪ 제조업체는 제조업체명으로 식별한다.
- ⑫ 회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.
- ⑬ 게시글에 대한 글번호, 글제목, 글내용, 작성일자 정보를 유지해야 한다.
- ⑭ 게시글은 글번호로 식별한다.

## 03 개념적 설계



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출한 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

## 03 개념적 설계



### ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출한 결과

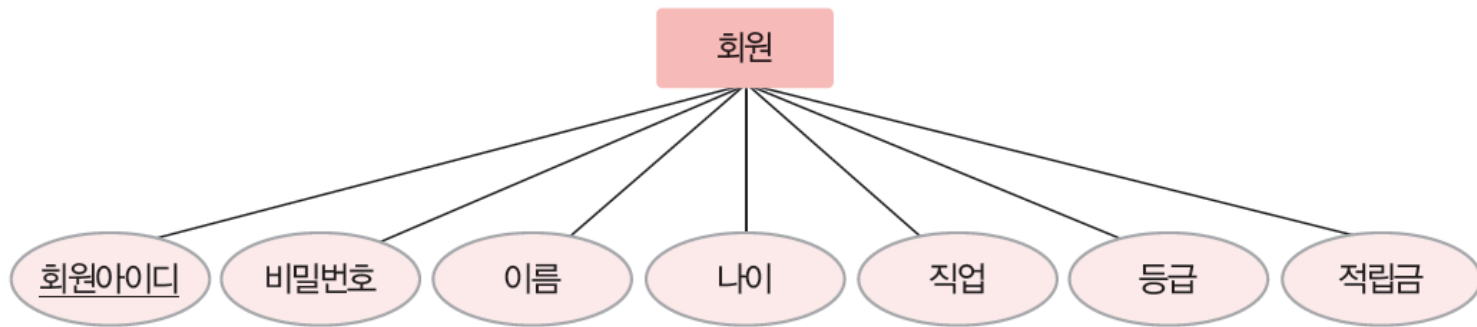


그림 8-13 회원 개체의 E-R 다이어그램

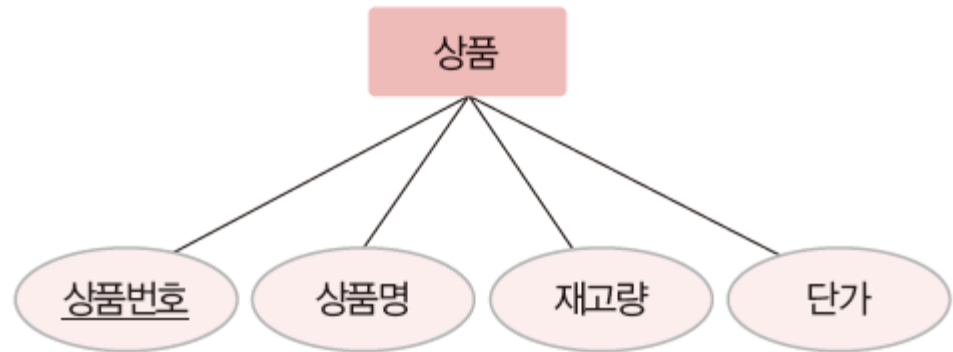


그림 8-14 상품 개체의 E-R 다이어그램

# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 1) 개체와 속성 추출 예

- [그림 8-4]의 요구 사항 명세서에서 개체와 속성을 추출한 결과

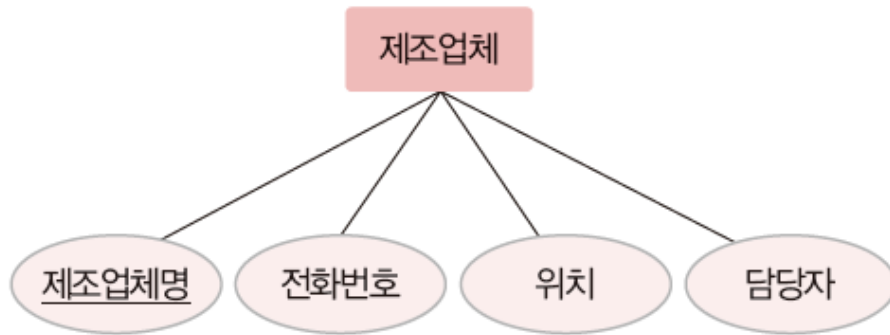


그림 8-15 제조업체 개체의 E-R 다이어그램

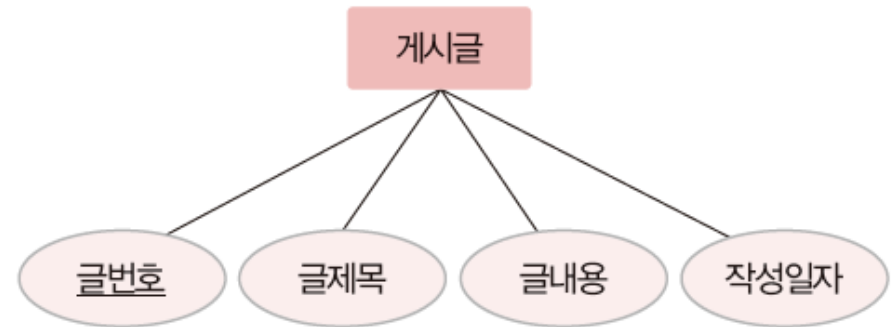


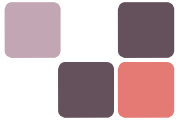
그림 8-16 게시글 개체의 E-R 다이어그램





### ❖ 개념적 설계 – (STEP 2) 관계 추출

- 관계 : 개체 간의 의미 있는 연관성
- 관계 추출 방법
  - 요구 사항 문장에서 개체 간의 연관성을 의미 있게 표현한 **동사**를 찾아라!
    - 의미가 같은 동사가 여러 개일 경우는 대표 동사 하나만 선택
  - 찾아낸 관계에 대해 매핑 카디널리티와 참여 특성을 결정하라!
    - 매핑 카디널리티 : 일대일(1:1), 일대다(1:n), 다대다(n:m)
    - 참여 특성 : 필수적 참여 / 선택적 참여



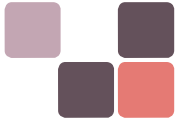
### ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출하는 과정

- 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)

- "입력해야 한다"는 개체와 개체의 관계를 표현하는 동사로 볼 수 없으므로 제외
- "부여된다"는 개체와 개체의 관계를 표현하는 동사로 볼 수 없으므로 제외
- "식별한다"는 개체와 개체의 관계를 표현하는 동사로 볼 수 없으므로 제외



### ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출하는 과정

- ⑥ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.

그림 8-18 요구 사항 문장에서 동사를 선별한 예 (B)

### [추출 결과]

- 관계: 주문
  - “회원” 개체와 “상품” 개체가 맺는 관계, 다대다(n:m) 관계
  - “회원” 개체는 관계에 선택적으로 참여 / “상품” 개체는 관계에 선택적으로 참여
- “주문” 관계의 속성 : 주문번호, 주문수량, 배송지, 주문일자



### ❖ 개념적 설계 – (STEP 2) 관계 추출 예

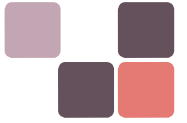
- [그림 8-4]의 요구 사항 명세서에서 관계를 추출하는 과정

- ⑧ 각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ⑨ 제조업체가 상품을 공급하면 공급일자와 공급량 정보를 유지해야 한다.

그림 8-19 요구 사항 문장에서 동사를 선별한 예 (C)

### [추출 결과]

- 관계: 공급
  - “상품” 개체와 “제조업체” 개체가 맺는 관계, 일대다(1:n) 관계
  - “상품” 개체는 관계에 필수적으로 참여 / “제조업체” 개체는 관계에 선택적으로 참여
- “공급” 관계의 속성 : 공급일자, 공급량



### ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출하는 과정

⑫ 회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.

그림 8-20 요구 사항 문장에서 동사를 선별한 예 (D)

### [추출 결과]

- 관계: 작성
  - “회원” 개체와 “게시글” 개체가 맺는 관계, 일대다(1:n) 관계
  - “회원” 개체는 관계에 선택적으로 참여 / “게시글” 개체는 관계에 필수적으로 참여

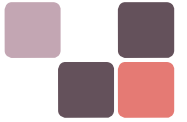


## ❖ 개념적 설계 – (STEP 2) 관계 추출 예

### ■ [그림 8-4]의 요구 사항 명세서에서 관계를 추출하는 과정

- ① 한빛 마트에 회원으로 가입하려면 회원아이디, 비밀번호, 이름, 나이, 직업을 입력해야 한다.
- ② 가입한 회원에게는 등급과 적립금이 부여된다.
- ③ 회원은 회원아이디로 식별한다.
- ④ 상품에 대한 상품번호, 상품명, 재고량, 단가 정보를 유지해야 한다.
- ⑤ 상품은 상품번호로 식별한다.
- ⑥ 회원은 여러 상품을 주문할 수 있고, 하나의 상품을 여러 회원이 주문할 수 있다.
- ⑦ 회원이 상품을 주문하면 주문에 대한 주문번호, 주문수량, 배송지, 주문일자 정보를 유지해야 한다.
- ⑧ 각 상품은 한 제조업체가 공급하고, 제조업체 하나는 여러 상품을 공급할 수 있다.
- ⑨ 제조업체가 상품을 공급하면 공급일자와 공급량 정보를 유지해야 한다.
- ⑩ 제조업체에 대한 제조업체명, 전화번호, 위치, 담당자 정보를 유지해야 한다.
- ⑪ 제조업체는 제조업체명으로 식별한다.
- ⑫ 회원은 게시글을 여러 개 작성할 수 있고, 게시글 하나는 한 명의 회원만 작성할 수 있다.
- ⑬ 게시글에 대한 글번호, 글제목, 글내용, 작성일자 정보를 유지해야 한다.
- ⑭ 게시글은 글번호로 식별한다.

# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출한 결과

관계	관계에 참여하는 개체	관계 유형	속성
주문	회원(선택) 상품(선택)	다대다	주문번호, 주문수량, 배송지, 주문일자
공급	상품(필수) 제조업체(선택)	일대다	공급일자, 공급량
작성	회원(선택) 게시글(필수)	일대다	

그림 8-22 한빛 마트 요구 사항 명세서에서 관계와 관계의 속성을 추출한 최종 결과

# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출한 결과

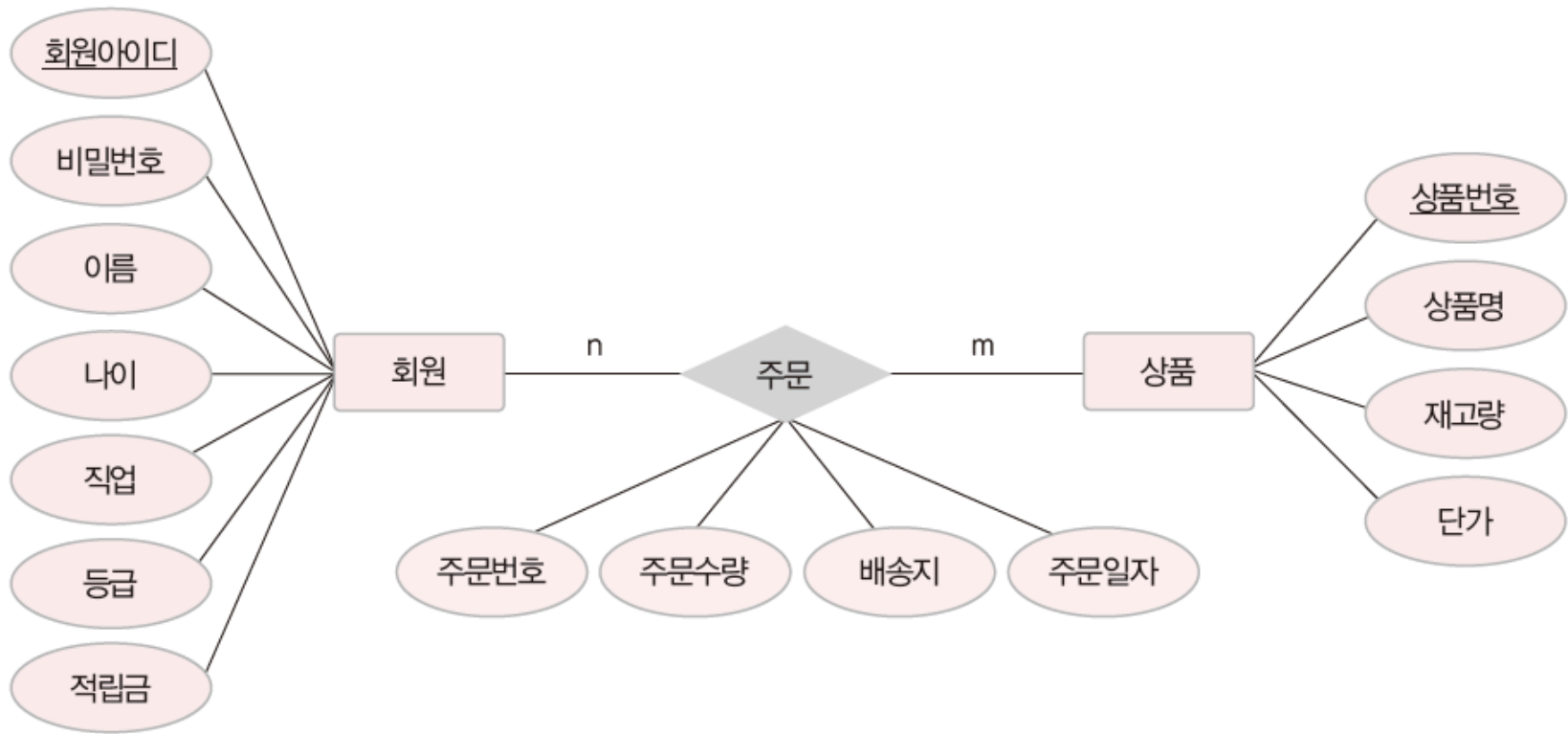


그림 8-23 주문 관계의 E-R 다이어그램



# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- [그림 8-4]의 요구 사항 명세서에서 관계를 추출한 결과

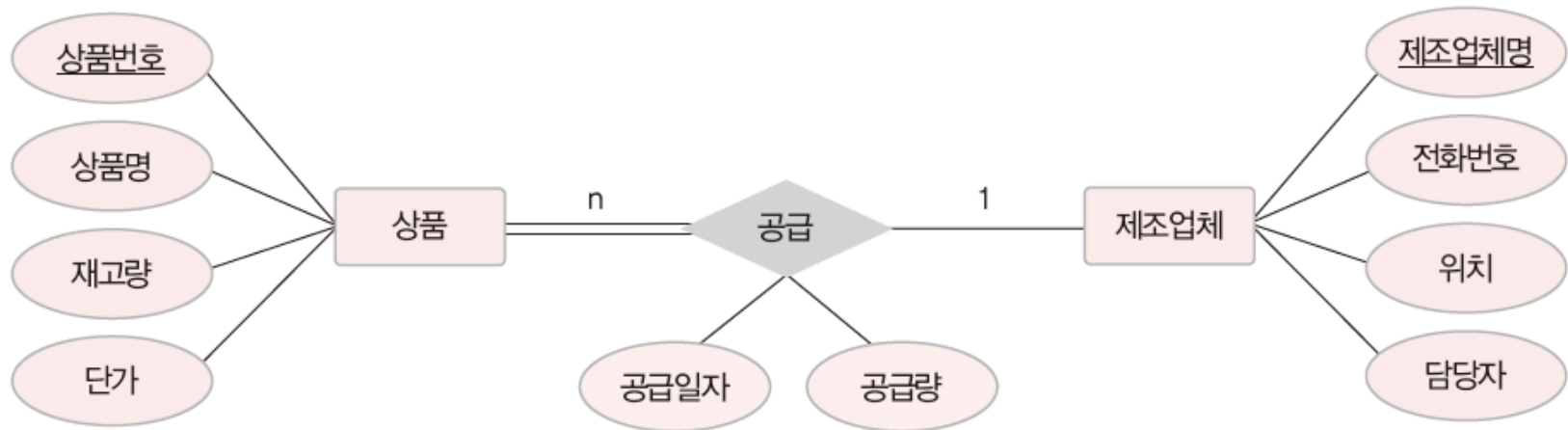


그림 8-24 공급 관계의 E-R 다이어그램

# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 2) 관계 추출 예

- 8페이지 [그림 8-4]의 요구 사항 명세서에서 관계를 추출한 결과

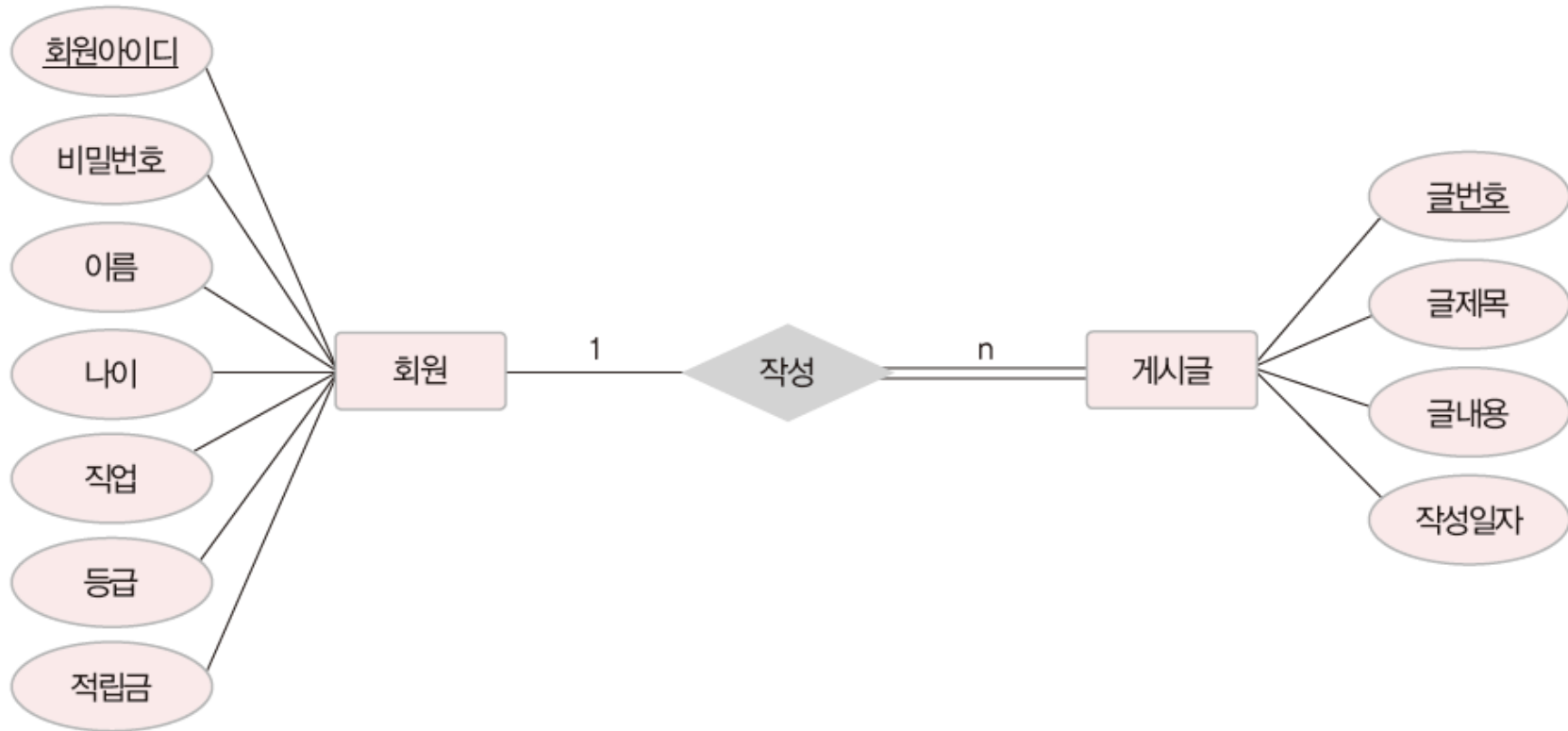
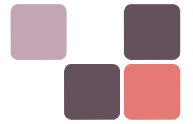


그림 8-25 작성 관계의 E-R 다이어그램

# 03 개념적 설계



## ❖ 개념적 설계 – (STEP 3) E-R 다이어그램 작성

- [그림 8-4]의 요구 사항 명세서를 개념적 스키마로 작성한 결과

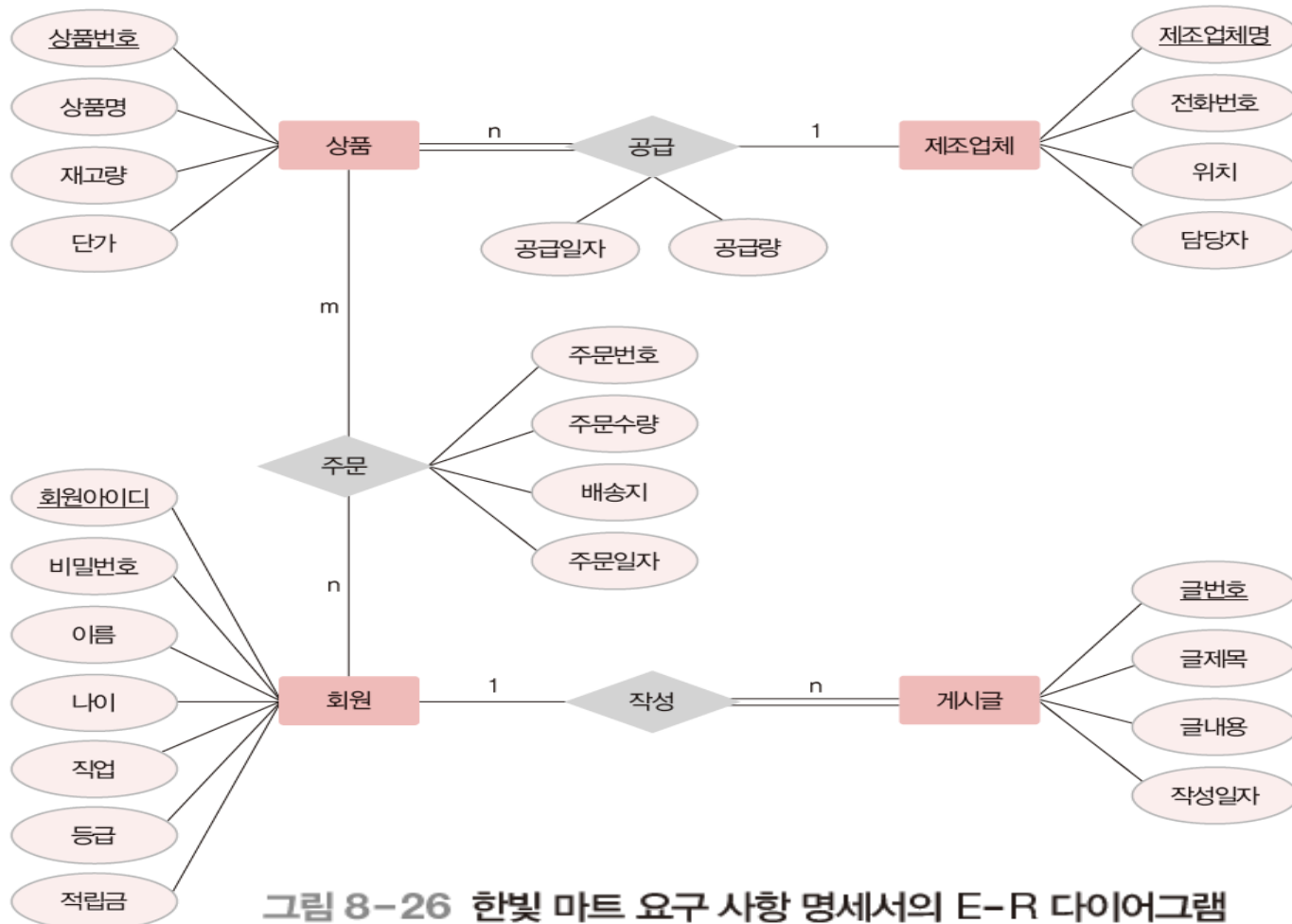


그림 8-26 한빛 마트 요구 사항 명세서의 E-R 다이어그램

## 03 개념적 설계

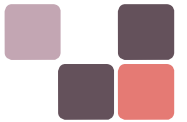


그림 8-5 데이터베이스 설계 : 개념적 설계 단계



## ❖ 설계 3 단계 : 논리적 설계

### ■ 목적

- DBMS에 적합한 논리적 스키마 설계
- 개념적 스키마를 논리적 데이터 모델을 이용해 논리적 구조로 표현
  - ➔ 논리적 모델링(데이터 모델링)
  - 일반적으로 관계 데이터 모델을 많이 이용

### ■ 결과물

- 논리적 스키마 : 릴레이션 스키마

### ■ 주요 작업

- 개념적 설계 단계의 결과물인 E-R 다이어그램을 릴레이션 스키마로 변환
- 릴레이션 스키마로 변환 후 속성의 데이터 타입, 길이, 널 값 허용 여부, 기본 값, 제약조건 등을 세부적으로 결정하고 결과를 문서화시킴



## ❖ 설계 3 단계 : 논리적 설계

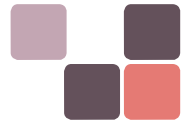
- E-R 다이어그램을 릴레이션 스키마로 변환하는 규칙
  - 규칙 1 : 모든 개체는 릴레이션으로 변환한다.
  - 규칙 2 : 다대다( $n:m$ ) 관계는 릴레이션으로 변환한다.
  - 규칙 3 : 일대다( $1:n$ ) 관계는 외래키로 표현한다.
  - 규칙 4 : 일대일( $1:1$ ) 관계는 외래키로 표현한다.
  - 규칙 5 : 다중 값 속성은 릴레이션으로 변환한다.
- 변환 규칙을 순서대로 적용하되, 해당되지 않는 규칙은 제외함



### ❖ 논리적 설계 – (규칙 1) 모든 개체는 릴레이션으로 변환한다

- E-R 다이어그램의 각 개체를 하나의 릴레이션으로 변환
  - 개체의 이름 → 릴레이션 이름
  - 개체의 속성 → 릴레이션의 속성
  - 개체의 키 속성 → 릴레이션의 기본키
  - 개체의 속성이 복합 속성인 경우에는 복합 속성을 구성하고 있는 단순 속성만 릴레이션의 속성으로 변환

# 04 논리적 설계



## ❖ 논리적 설계 – (규칙 1) 모든 개체는 릴레이션으로 변환한다

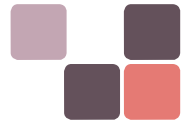


그림 8-28 개체를 릴레이션으로 변환하는 규칙을 적용한 예

상품(상품번호, 상품명, 재고량, 단가)



## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 1) 모든 개체는 릴레이션으로 변환한다

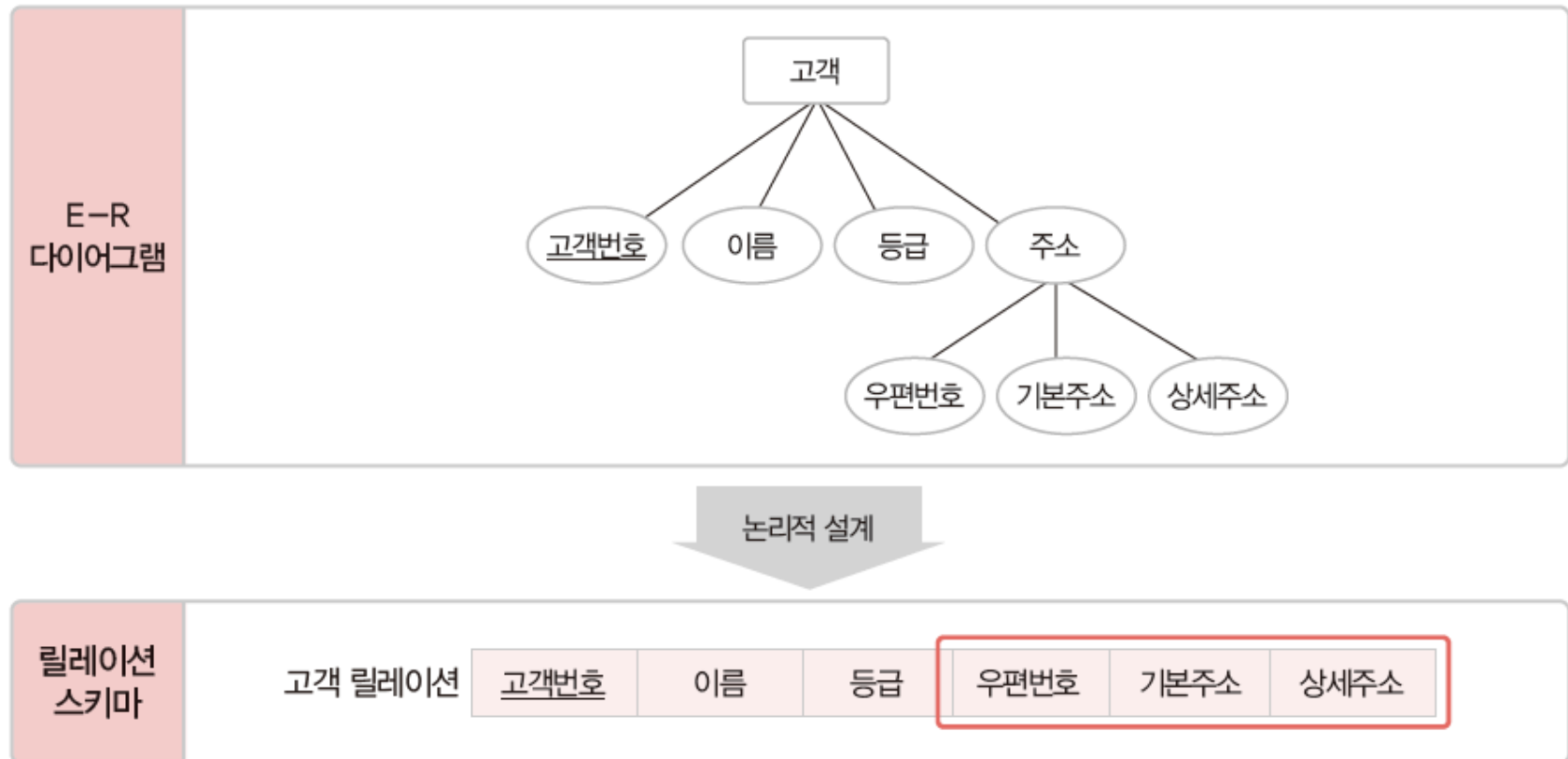


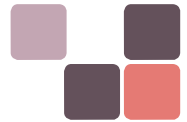
그림 8-29 복합 속성을 가지는 개체를 릴레이션으로 변환하는 예



### ❖ 논리적 설계 – (규칙 2) 다대다 관계는 릴레이션으로 변환한다

- E-R 다이어그램의 다대다 관계를 하나의 릴레이션으로 변환
  - 관계의 이름 → 릴레이션 이름
  - 관계의 속성 → 릴레이션의 속성
  - 관계에 참여하는 개체를 규칙 1에 따라 릴레이션으로 변환한 후 이 릴레이션의 기본키를 관계 릴레이션에 포함시켜 외래키로 지정하고, 외래키들을 조합하여 관계 릴레이션의 기본키로 지정

# 04 논리적 설계



## ❖ 논리적 설계 – (규칙 2) 다대다 관계는 릴레이션으로 변환한다

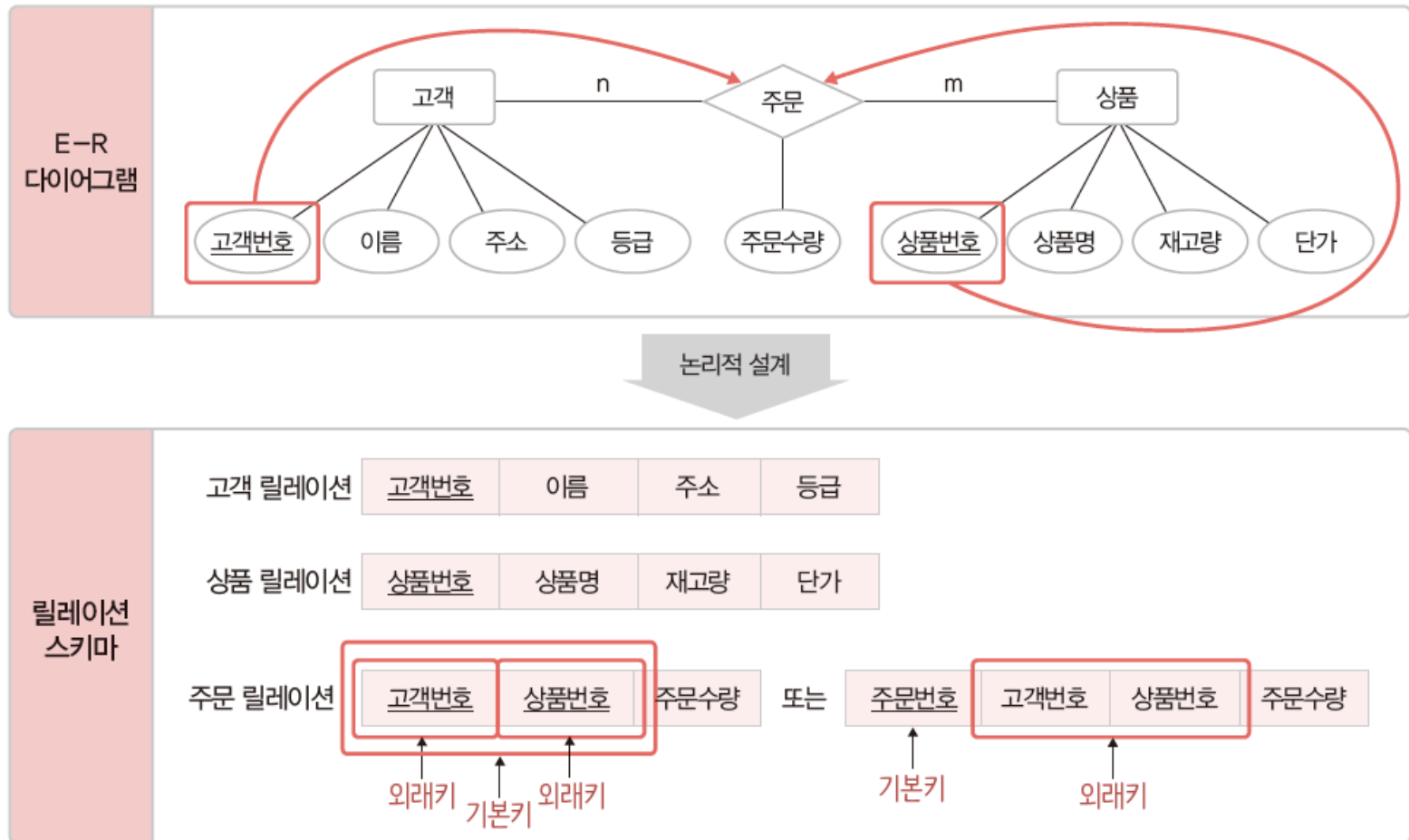
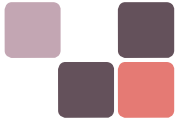


그림 8-30 다대다 관계를 릴레이션으로 변환하는 규칙을 적용한 예



### ❖ 논리적 설계 – (규칙 3) 일대다 관계는 외래키로 표현한다

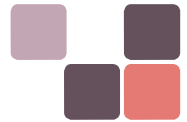
- E-R 다이어그램의 일대다 관계는 외래키로만 표현
  - (규칙 3-1) 일반적인 일대다 관계는 외래키로 표현한다.
  - (규칙 3-2) 약한 개체가 참여하는 일대다 관계는 외래키를 포함해서 기본키로 지정한다.



### ❖ 논리적 설계 – (규칙 3) 일대다 관계는 외래키로 표현한다

- (규칙 3-1) 일반적인 일대다 관계는 외래키로 표현한다.
  - 일대다(1:n) 관계에서 1측 개체 릴레이션의 기본키를 n측 개체 릴레이션에 포함시켜 외래키로 지정
  - 관계의 속성들도 n측 개체 릴레이션에 포함시킴

## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 3) 일대다 관계는 외래키로 표현한다

- (규칙 3-1) 일반적인 일대다 관계는 외래키로 표현한다.



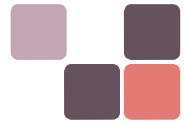
그림 8-31 일반적인 개체가 참여하는 일대다 관계를 외래키로 표현하는 규칙을 적용한 예



### ❖ 논리적 설계 – (규칙 3) 일대다 관계는 외래키로 표현한다

- (규칙 3-2) 약한 개체가 참여하는 일대다 관계는 외래키를 포함해서 기본키를 지정한다.
  - 일대다(1:n) 관계에서 1측 개체 릴레이션의 기본키를 n측 개체 릴레이션에 포함시켜 외래키로 지정
  - 관계의 속성들도 n측 개체 릴레이션에 포함시킴
  - n측 개체 릴레이션은 외래키를 포함하여 기본키를 지정함
    - 약한 개체는 강한 개체에 따라 존재 여부가 결정되므로 강한 개체의 기본키를 이용해 식별해야 함

## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 3) 일대다 관계는 외래키로 표현한다

- (규칙 3-2) 약한 개체가 참여하는 일대다 관계는 외래키를 포함해서 기본키를 지정한다.

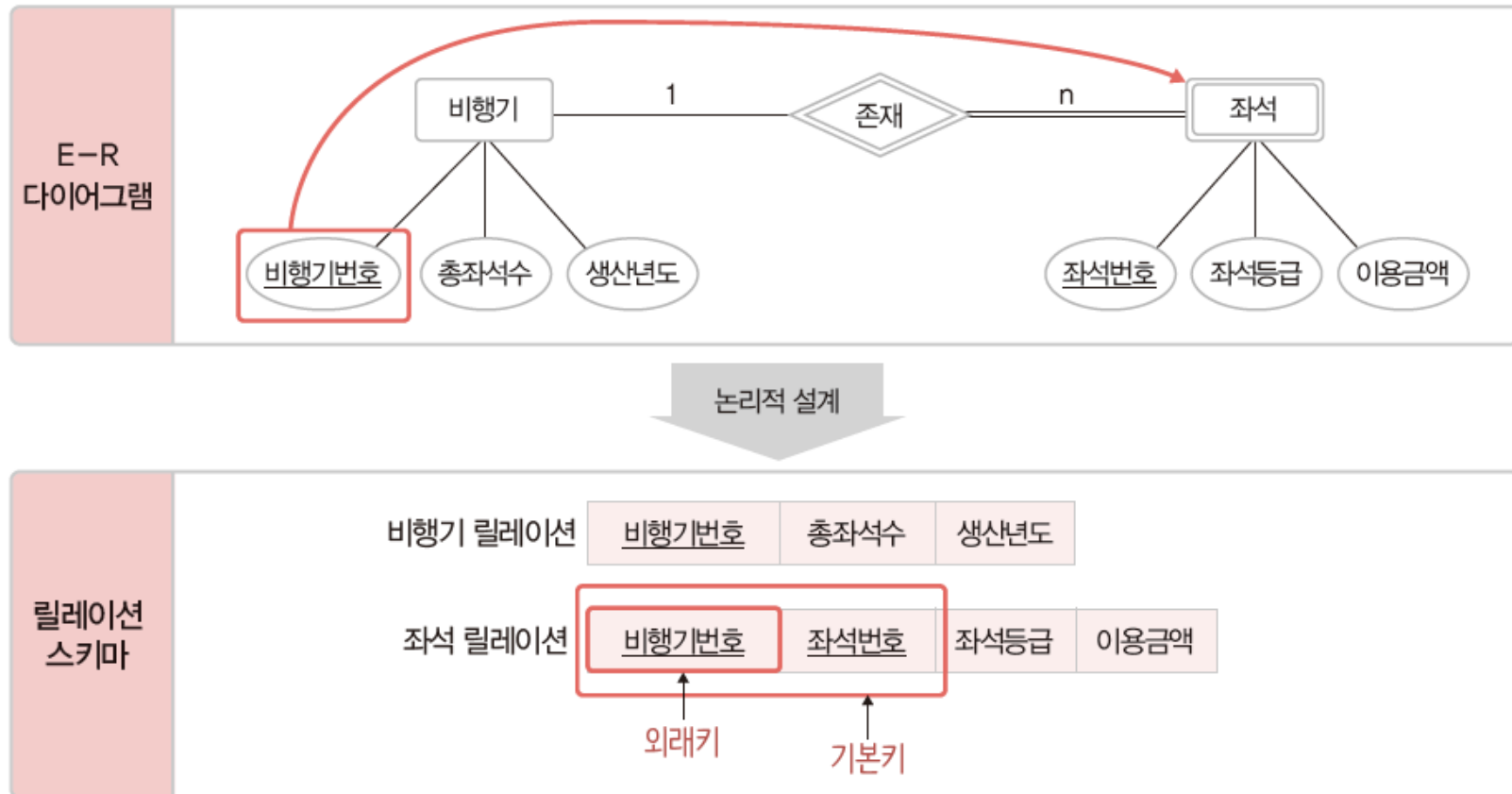


그림 8-32 약한 개체가 참여하는 일대다 관계를 외래키로 표현하는 규칙을 적용한 예





### ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다

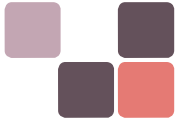
- E-R 다이어그램의 일대일 관계는 외래키로만 표현
  - (규칙 4-1) 일반적인 일대일 관계는 외래키를 서로 주고받는다.
  - (규칙 4-2) 일대일 관계에 필수적으로 참여하는 개체의 릴레이션만 외래키를 받는다.
  - (규칙 4-3) 모든 개체가 일대일 관계에 필수적으로 참여하면 릴레이션 하나로 합친다.



### ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다

- (규칙 4-1) 일반적인 일대일 관계는 외래키를 서로 주고받는다.
  - 관계에 참여하는 개체 릴레이션들이 서로의 기본키를 주고받아 외래키로 지정
  - 관계의 속성들도 모든 개체 릴레이션에 포함시킴
  - 불필요한 데이터 중복이 발생할 수 있음

## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다

- (규칙 4-1) 일반적인 일대일 관계는 외래키를 서로 주고받는다.

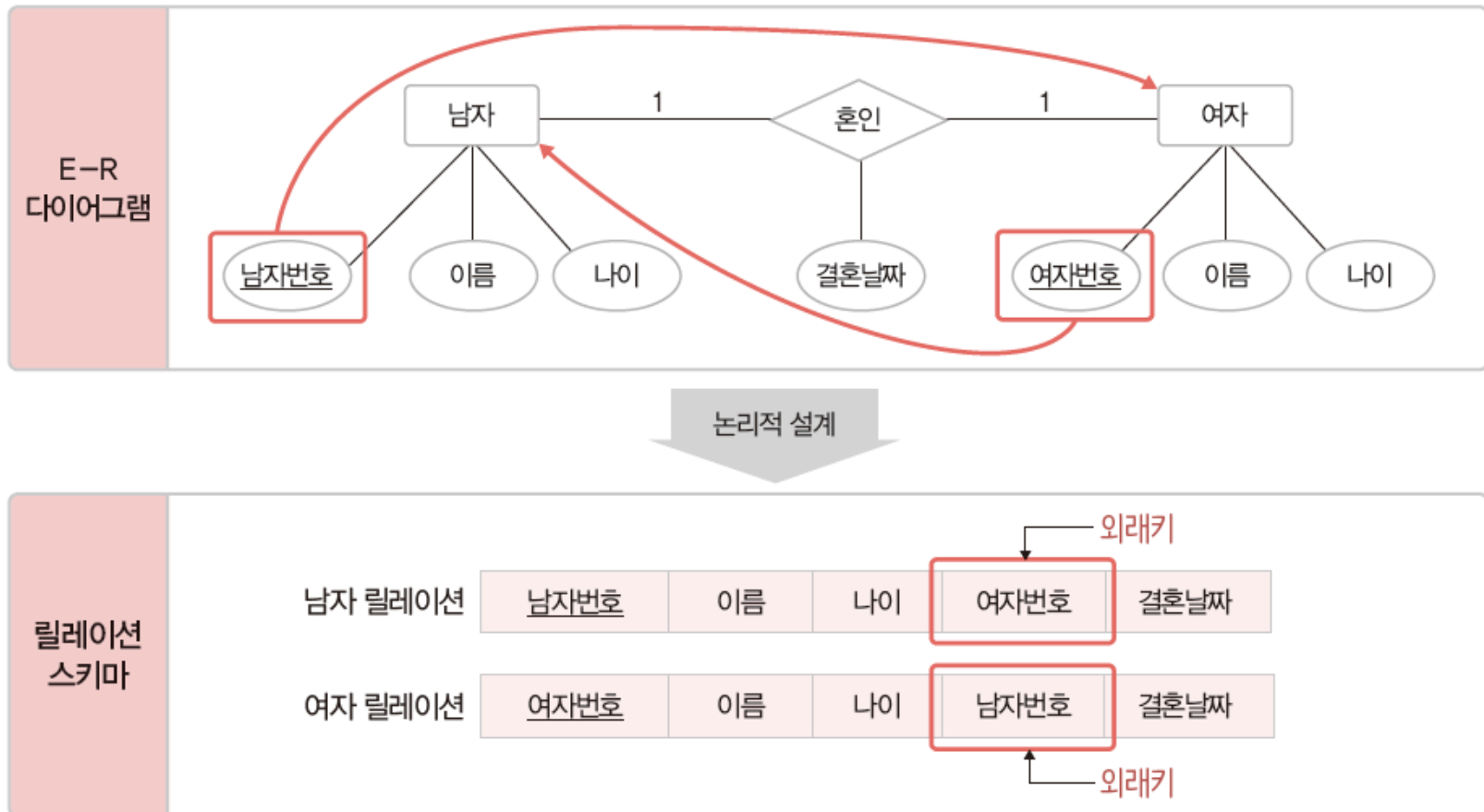


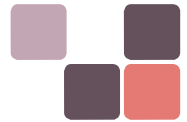
그림 8-33 일반적인 일대일 관계를 외래키로 표현하는 규칙을 적용한 예



### ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다

- (규칙 4-2) 필수적으로 참여하는 개체 릴레이션만 외래키를 받는다.
  - 관계에 필수적으로 참여하는 개체 릴레이션에만 외래키를 포함시킴
  - 관계의 속성들은 관계에 필수적으로 참여하는 개체 릴레이션에 포함시킴

# 04 논리적 설계



- ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다
  - (규칙 4-2) 필수적으로 참여하는 개체 릴레이션만 외래키를 받는다.

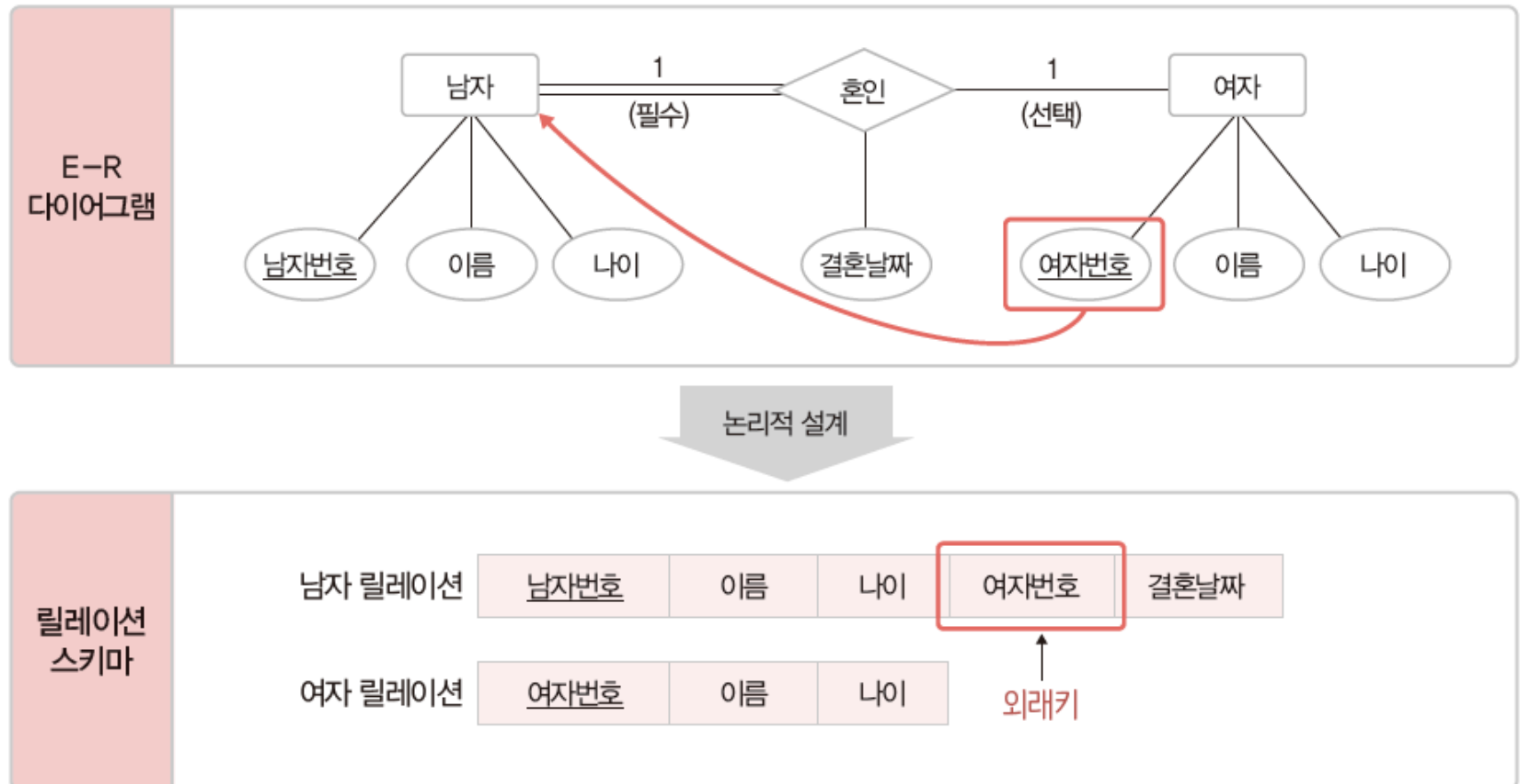


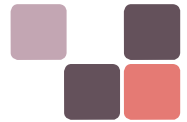
그림 8-34 일대일 관계에 필수적으로 참여하는 개체의 릴레이션이 외래키를 가지는 예



### ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다

- (규칙 4-3) 모든 개체가 필수적으로 참여하면 릴레이션 하나로 합친다.
  - 관계에 참여하는 개체 릴레이션들을 하나의 릴레이션으로 합쳐서 표현
  - 관계의 이름을 릴레이션 이름으로 사용하고, 관계에 참여하는 두 개체의 속성들을 관계 릴레이션에 모두 포함시킴
  - 두 개체 릴레이션의 키 속성을 조합하여 관계 릴레이션의 기본키로 지정

## 04 논리적 설계



- ❖ 논리적 설계 – (규칙 4) 일대일 관계는 외래키로 표현한다
  - (규칙 4-3) 모든 개체가 필수적으로 참여하면 릴레이션 하나로 합친다.

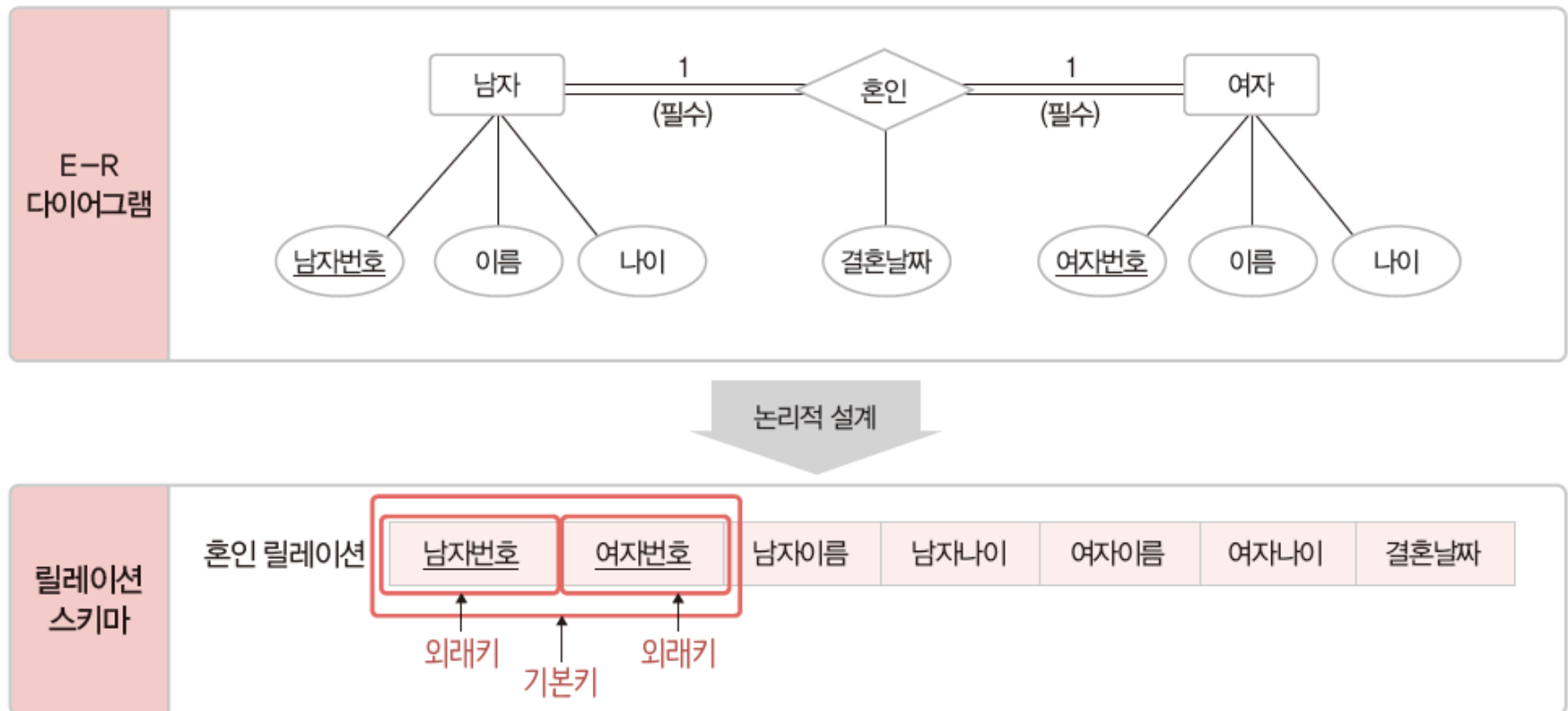


그림 8-35 일대일 관계에 모든 개체가 필수적으로 참여하면 릴레이션을 통합하는 예

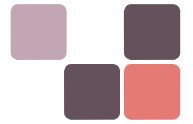


### ❖ 논리적 설계 – (규칙 5) 다중 값 속성은 릴레이션으로 변환한다

- E-R 다이어그램의 다중 값 속성은 독립적인 릴레이션으로 변환
  - 다중 값 속성과 함께 그 속성을 가지고 있던 개체 릴레이션의 기본키를 외래키로 가져와 새로운 릴레이션에 포함시킴
  - 새로운 릴레이션의 기본키는 다중 값 속성과 외래키를 조합하여 지정



## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 5) 다중 값 속성은 릴레이션으로 변환한다

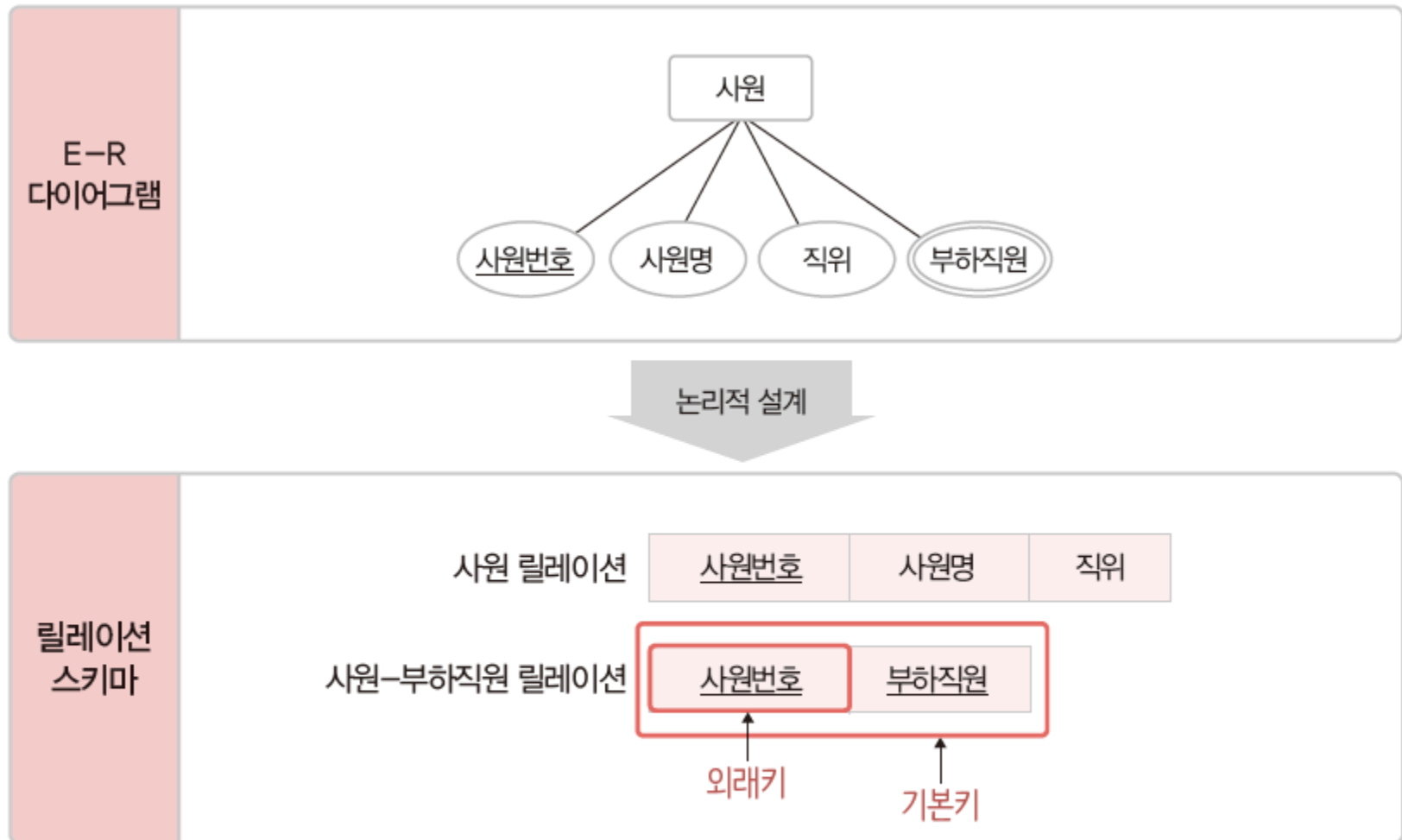
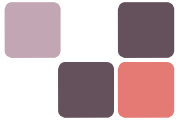


그림 8-36 다중 값 속성을 릴레이션으로 변환하는 규칙을 적용한 예

## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 5) 다중 값 속성은 릴레이션으로 변환한다

<u>사원번호</u>	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

사원 릴레이션은 "속성에 다중 값을 저장할 수 없다"는 릴레이션 특성을 위반함

## 04 논리적 설계



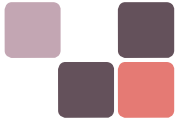
### ❖ 논리적 설계 – (규칙 5) 다중 값 속성은 릴레이션으로 변환한다

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

그림 8-38 릴레이션 특성에 맞게 부하직원 속성을 포함하는 사원 릴레이션

사원 릴레이션은 릴레이션 특성을 위반하지는 않지만  
사원번호, 사원명, 직위 속성의 값이 불필요하게 중복 저장되는 문제가 발생함

## 04 논리적 설계



### ❖ 논리적 설계 – (규칙 5) 다중 값 속성은 릴레이션으로 변환한다

사원 릴레이션			사원-부하직원 릴레이션	
<u>사원번호</u>	사원명	직위	<u>사원번호</u>	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

(규칙 5)에 따라 다중 값 속성을 독립적인 릴레이션으로 변환하면 불필요한 중복을 제거하면서도 릴레이션의 특성을 만족시킬 수 있다.



## ❖ 논리적 설계 – 기타 고려 사항

- 모든 관계를 독립적인 릴레이션으로 변환할 수 있다.
- 속성이 많은 관계는 유형에 상관없이 릴레이션으로의 변환을 고려할 수 있음

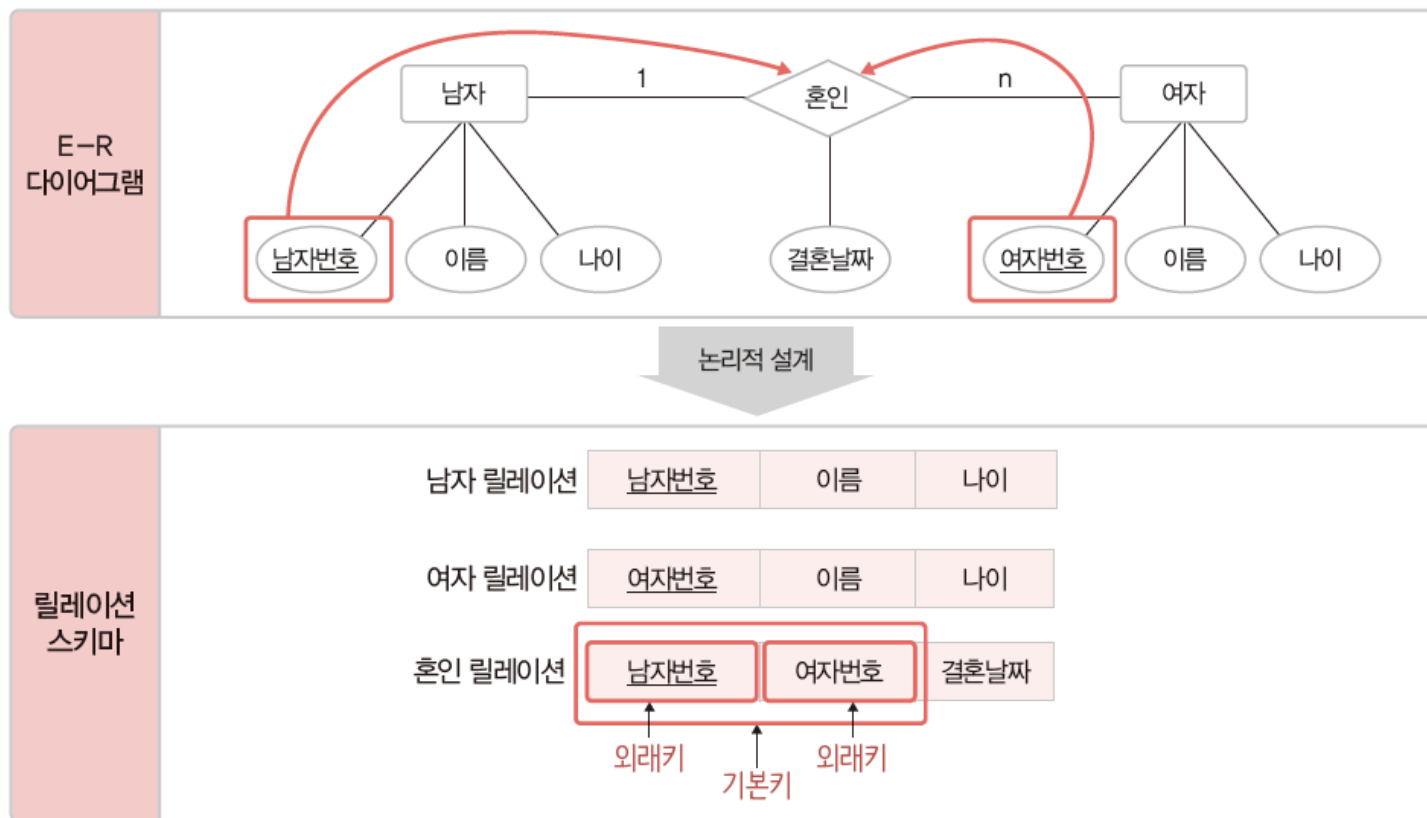
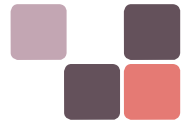


그림 8-40 일대일 관계를 릴레이션으로 변환하는 예



## ❖ 논리적 설계 – 기타 고려 사항

- 개체가 자기 자신과 관계를 맺는 순환 관계도 기본 규칙을 그대로 적용

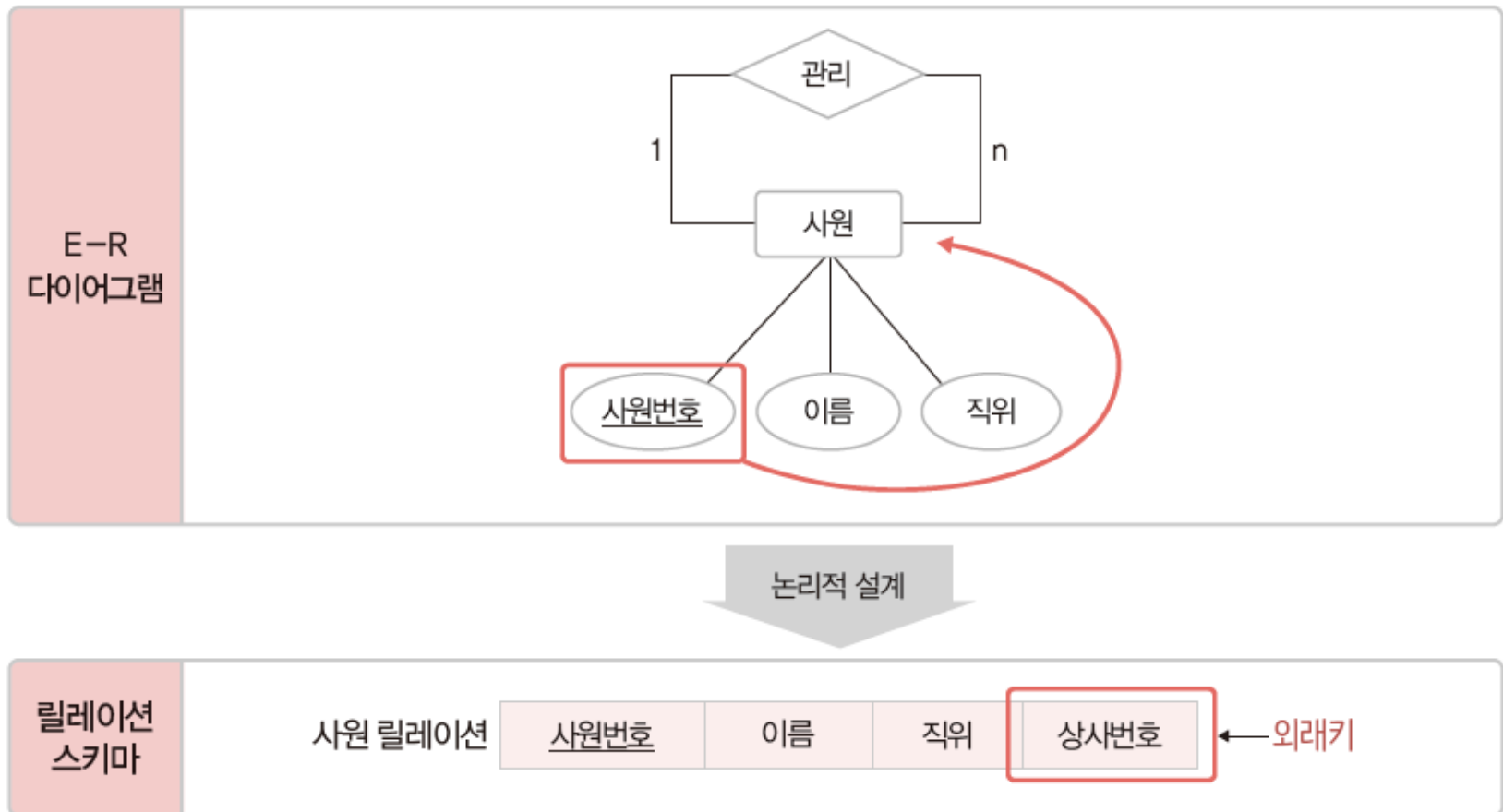
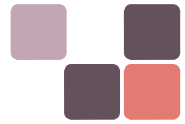


그림 8-41 순환 관계를 변환하는 예

# 04 논리적 설계



## ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

- [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- STEP 1) 규칙 1 적용

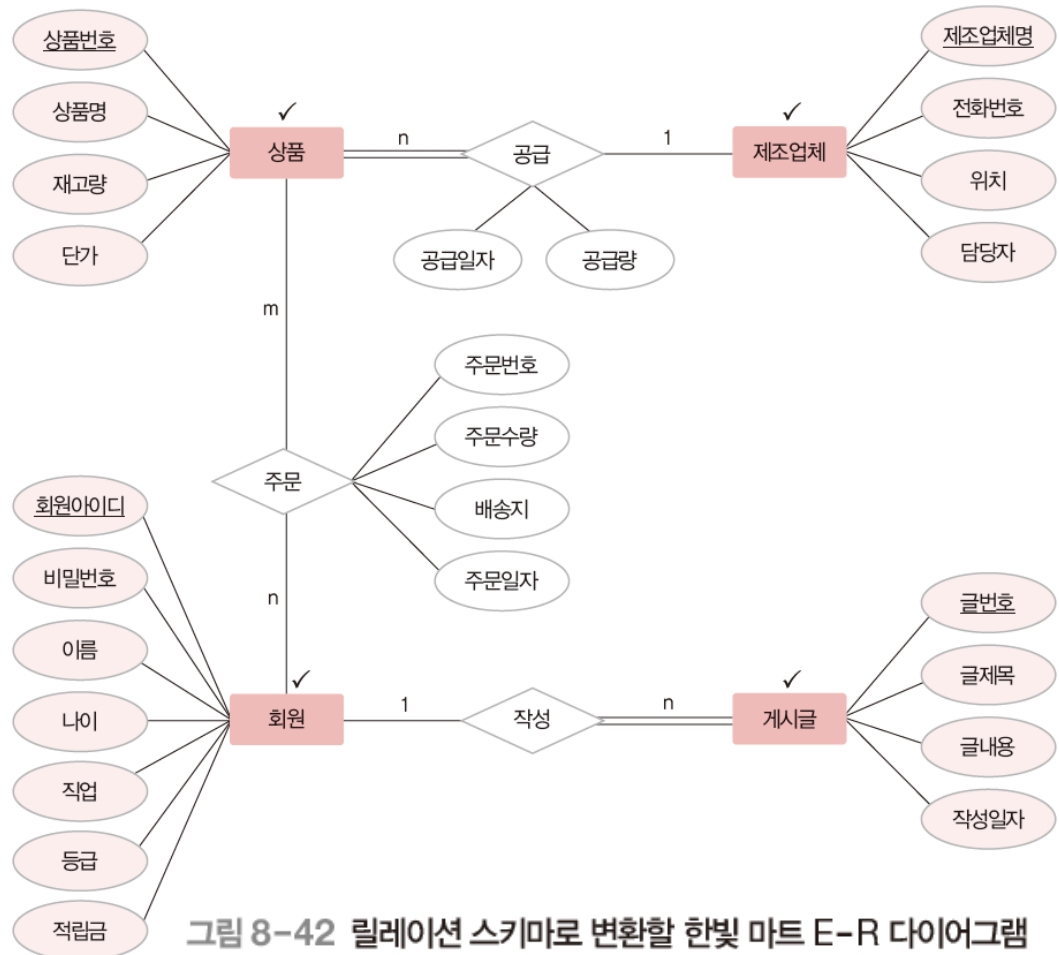


그림 8-42 릴레이션 스키마로 변환할 한빛 마트 E-R 다이어그램

## 04 논리적 설계



### ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

- [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- (규칙 1) 적용 결과

회원 릴레이션	<u>회원아이디</u>	비밀번호	이름	나이	직업	등급	적립금
---------	--------------	------	----	----	----	----	-----

상품 릴레이션	<u>상품번호</u>	상품명	재고량	단가
---------	-------------	-----	-----	----

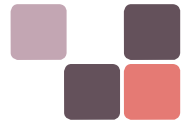
제조업체 릴레이션	<u>제조업체명</u>	전화번호	위치	담당자
-----------	--------------	------	----	-----

게시글 릴레이션	<u>글번호</u>	글제목	글내용	작성일자
----------	------------	-----	-----	------

그림 8-43 규칙 1을 적용한 결과

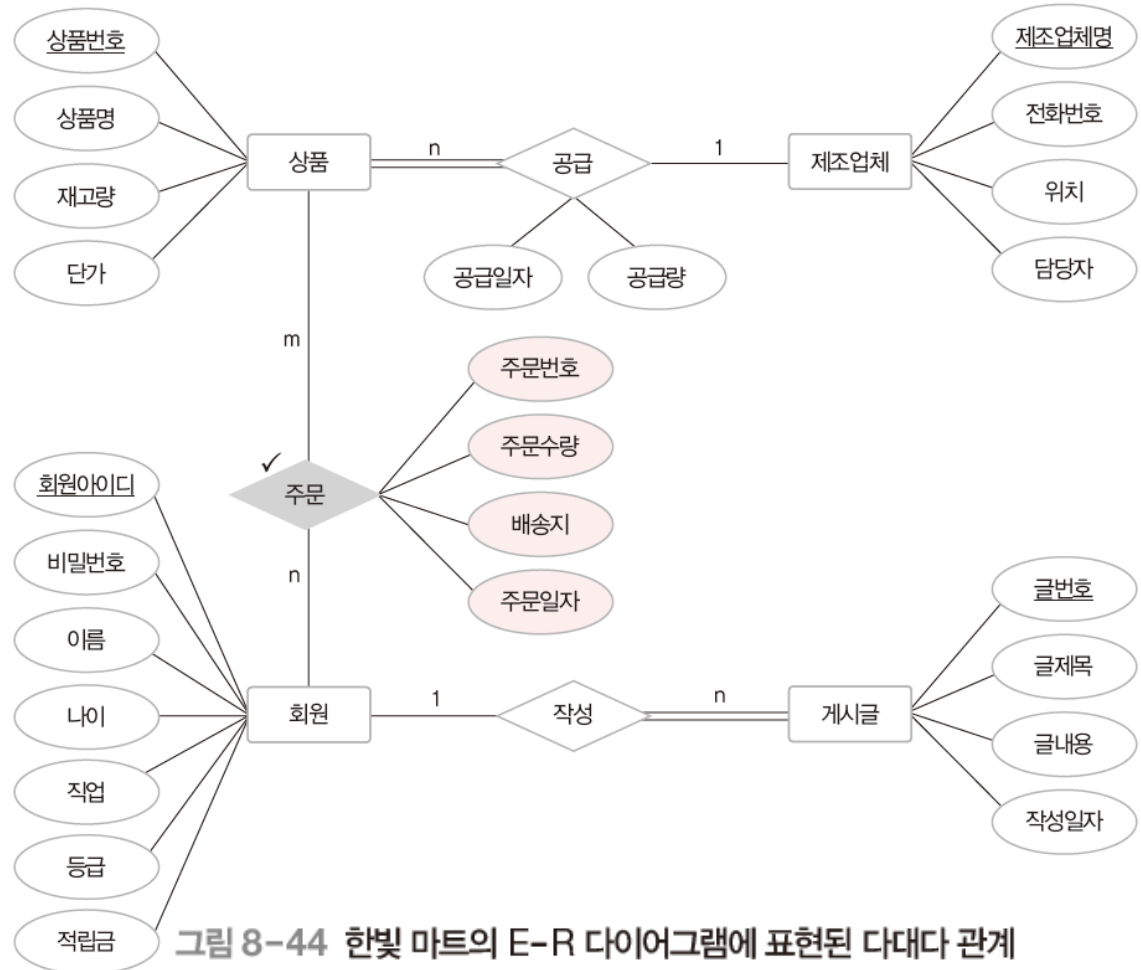


# 04 논리적 설계

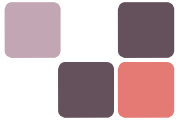


## ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

- [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- STEP 2) 규칙 2 적용



## 04 논리적 설계



### ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

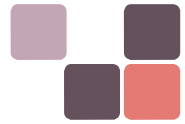
- 30페이지 [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- (규칙 2) 적용 결과

회원 릴레이션	<u>회원아이디</u>	비밀번호	이름	나이	직업	등급	적립금
상품 릴레이션	<u>상품번호</u>	상품명	재고량	단가			
제조업체 릴레이션	<u>제조업체명</u>	전화번호	위치	담당자			
게시글 릴레이션	<u>글번호</u>	글제목	글내용	작성일자			
주문 릴레이션	<u>회원아이디</u>	<u>상품번호</u>	주문번호	주문수량	배송지	주문일자	

외래키      기본키      외래키

그림 8-45 규칙 2를 적용한 결과

# 04 논리적 설계



## ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

■ [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정

■ STEP 3) 규칙 3 적용

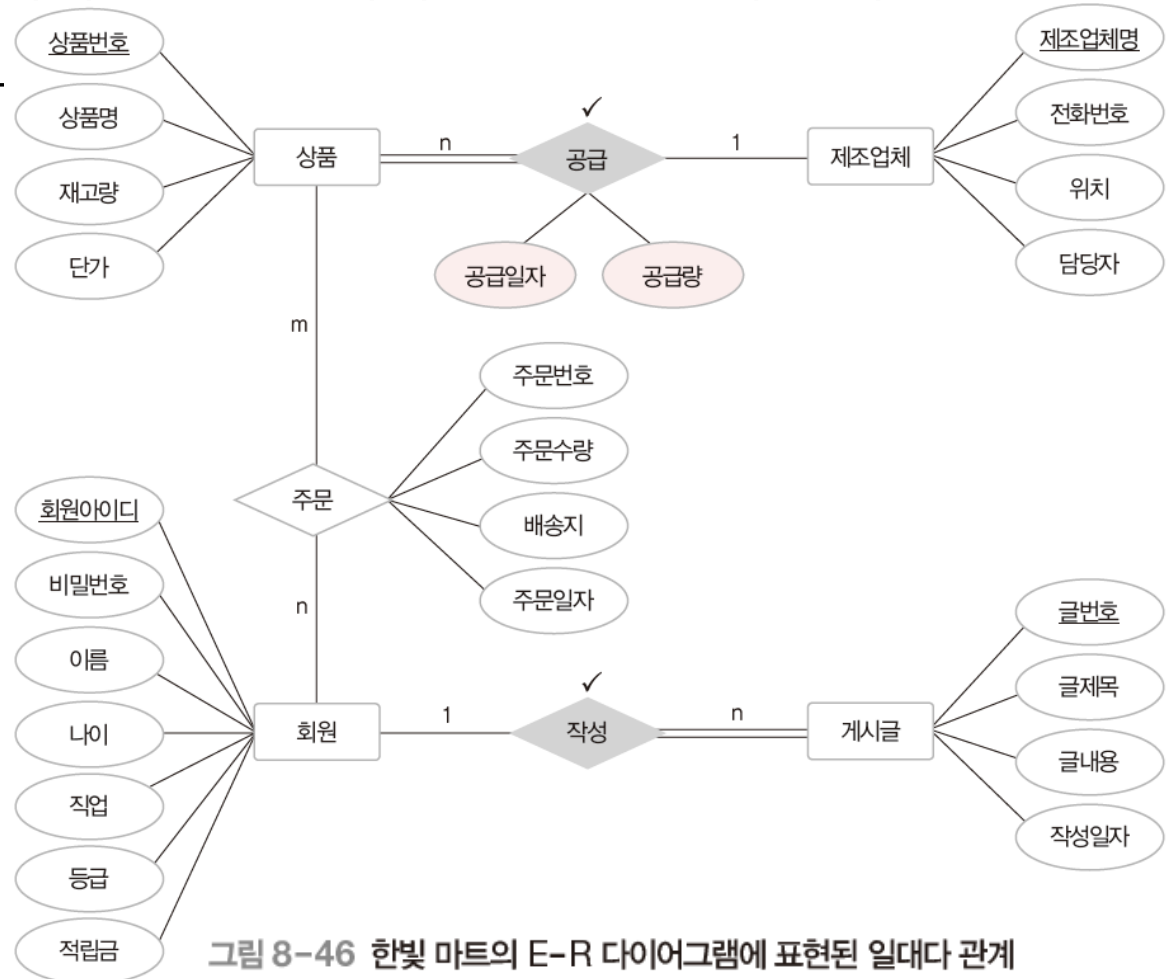
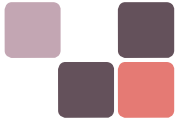


그림 8-46 한빛 마트의 E-R 다이어그램에 표현된 일대다 관계

## 04 논리적 설계



### ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

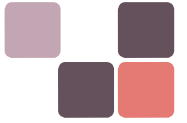
- 30페이지 [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- (규칙 3) 적용 결과

회원 릴레이션	<u>회원아이디</u>	비밀번호	이름	나이	직업	등급	적립금
상품 릴레이션	<u>상품번호</u>	상품명	재고량	단가	제조업체명	공급일자	공급량
제조업체 릴레이션	<u>제조업체명</u>	전화번호	위치	담당자			
게시글 릴레이션	<u>글번호</u>	글제목	글내용	작성일자	회원아이디		
주문 릴레이션	<u>회원아이디</u>	<u>상품번호</u>	주문번호	주문수량	배송지	주문일자	

외래키

외래키

그림 8-47 규칙 3을 적용한 결과



### ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

- [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환하는 과정
- STEP 4) 규칙 4 적용
  - 일대일 관계가 없으므로 규칙 4는 적용할 필요가 없음
- STEP 5) 규칙 5 적용
  - 다중 값 속성이 없으므로 규칙 5는 적용할 필요가 없음

## 04 논리적 설계



### ❖ 릴레이션 스키마 변환 규칙을 이용한 논리적 설계 예

- [그림 8-26]의 E-R 다이어그램을 릴레이션으로 변환한 최종 결과

회원 릴레이션	<u>회원아이디</u>	비밀번호	이름	나이	직업	등급	적립금
상품 릴레이션	<u>상품번호</u>	상품명	재고량	단가	제조업체명	공급일자	공급량
제조업체 릴레이션	<u>제조업체명</u>	전화번호	위치	담당자			
게시글 릴레이션	<u>글번호</u>	글제목	글내용	작성일자	회원아이디		
주문 릴레이션	<u>회원아이디</u>	<u>상품번호</u>	주문번호	주문수량	배송지	주문일자	

외래키

외래키

그림 8-47 규칙 3을 적용한 결과



## ❖ 논리적 설계 - 테이블 명세서 작성

- 릴레이션 스키마 변환 후 속성의 데이터 타입과 길이, 널 값 허용 여부, 기본값, 제약조건 등을 세부적으로 결정하고 문서화시킴
- 테이블 명세서 : 릴레이션 스키마에 대한 설계 정보를 기술한 문서

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				0이상
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

# 04 논리적 설계

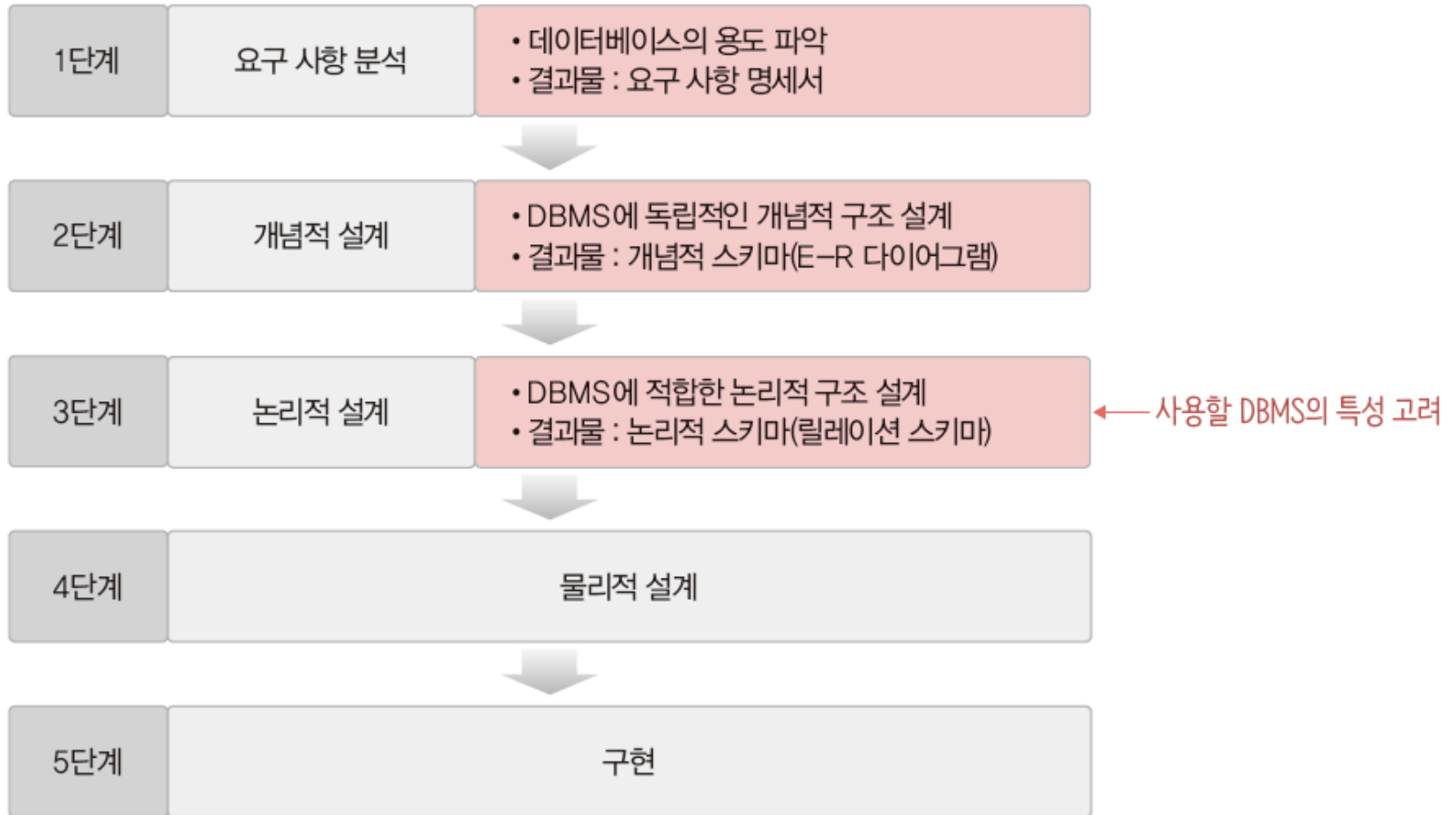


그림 8-27 데이터베이스 설계 : 논리적 설계 단계





## ❖ 설계 4 단계 : 물리적 설계

- 하드웨어나 운영체제의 특성을 고려하여 필요한 인덱스 구조나 내부 저장 구조 등에 대한 물리적인 구조를 설계

## ❖ 설계 5 단계 : 구현

- SQL로 작성한 명령문을 DBMS에서 실행하여 데이터베이스를 실제로 생성



## ❖ 테이블 명세서에 따라 SQL 문을 작성한 예

```
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

# 05 물리적 설계와 구현

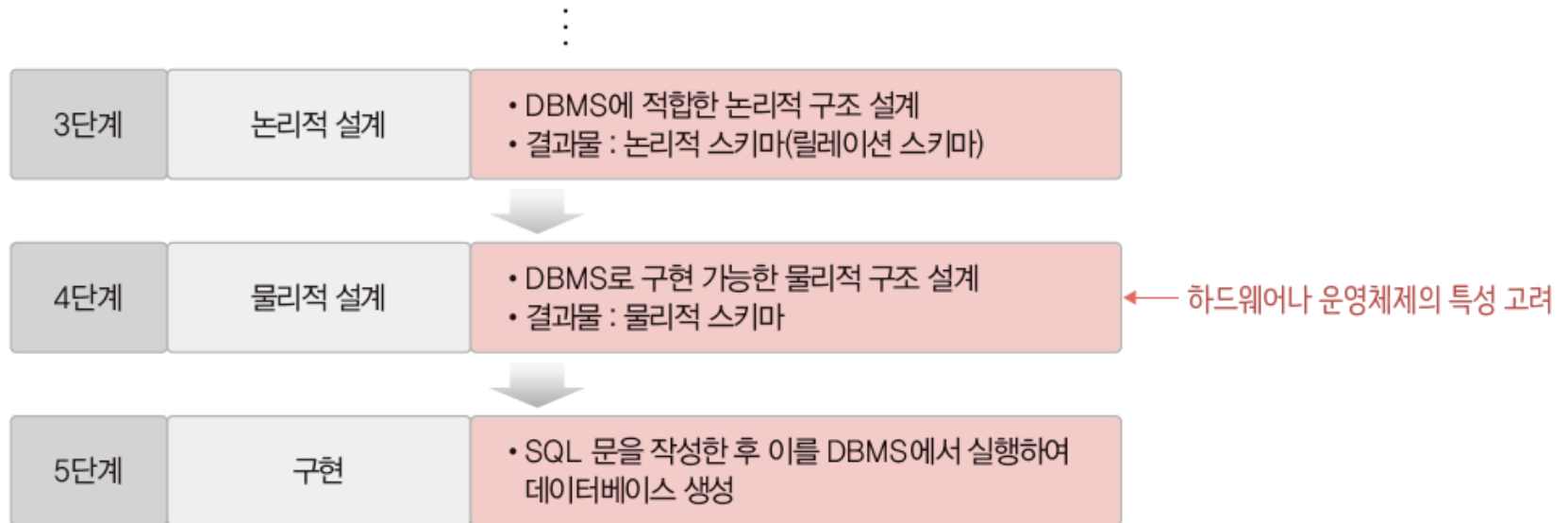
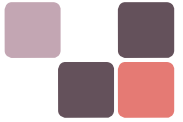


그림 8-49 데이터베이스 설계 : 물리적 설계와 구현