

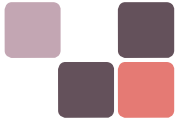
3주차.

데이터 모델링

- ✓ 논리적 데이터 모델 생성
- ✓ 데이터 모델링의 개념과 개체-관계 모델을 이용한 모델링

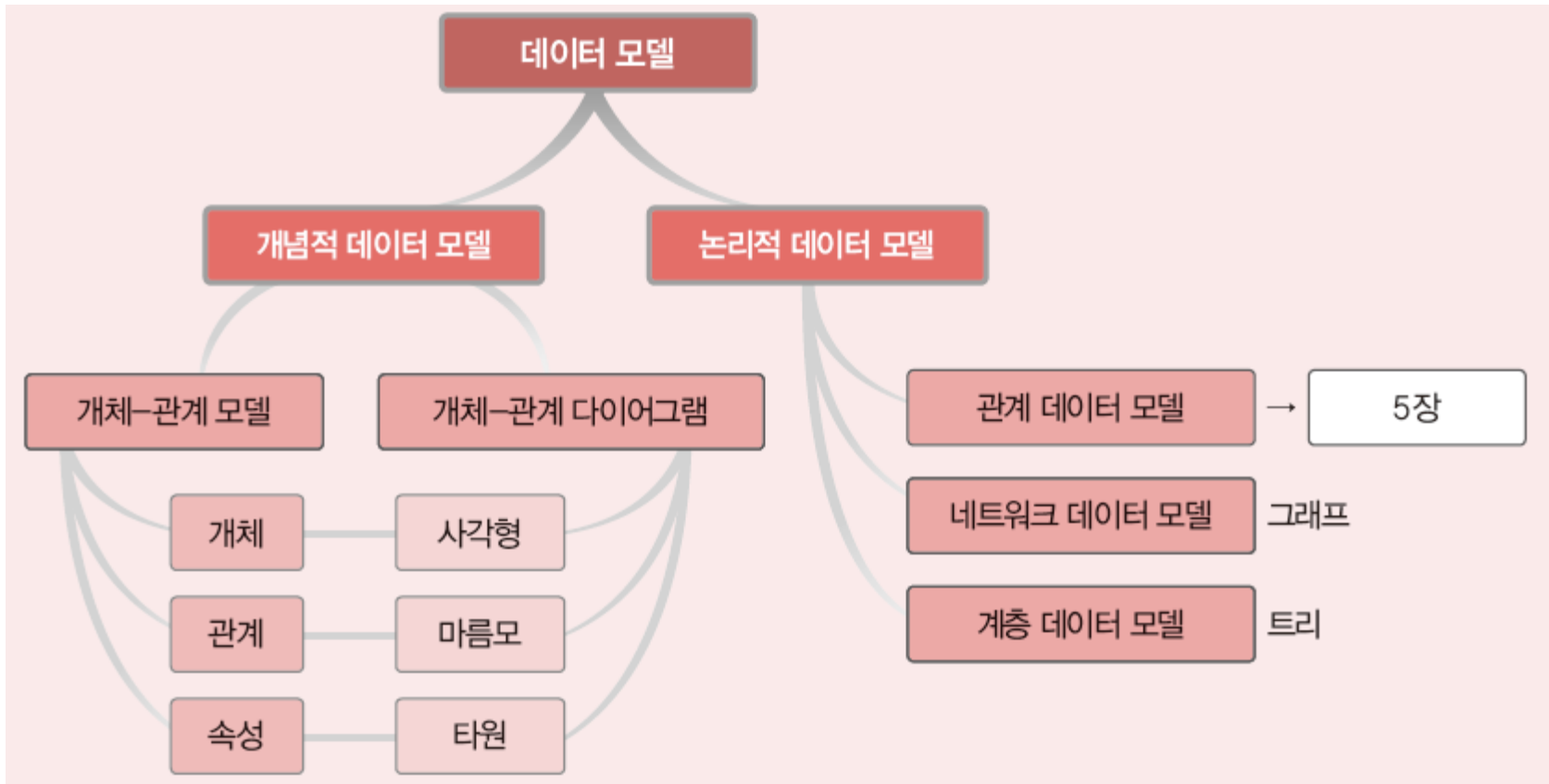
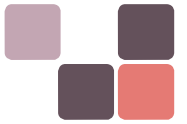
【Contents】

Ch04. 데이터 모델링



4장. 데이터 모델링

- 데이터 모델링과 데이터 모델의 개념
- 개체-관계 모델
- 논리적 데이터 모델



- ❖ 데이터 모델링과 데이터 모델의 개념을 이해한다.
- ❖ 개념적 데이터 모델인 개체 - 관계 모델을 이용해 모델링을 하는 방법을 익힌다.
- ❖ 개체 - 관계 모델을 개체 - 관계 다이어그램으로 작성하는 방법을 익힌다.
- ❖ 논리적 데이터 모델의 종류와 특징을 이해한다.

01 데이터 모델링과 데이터 모델의 개념

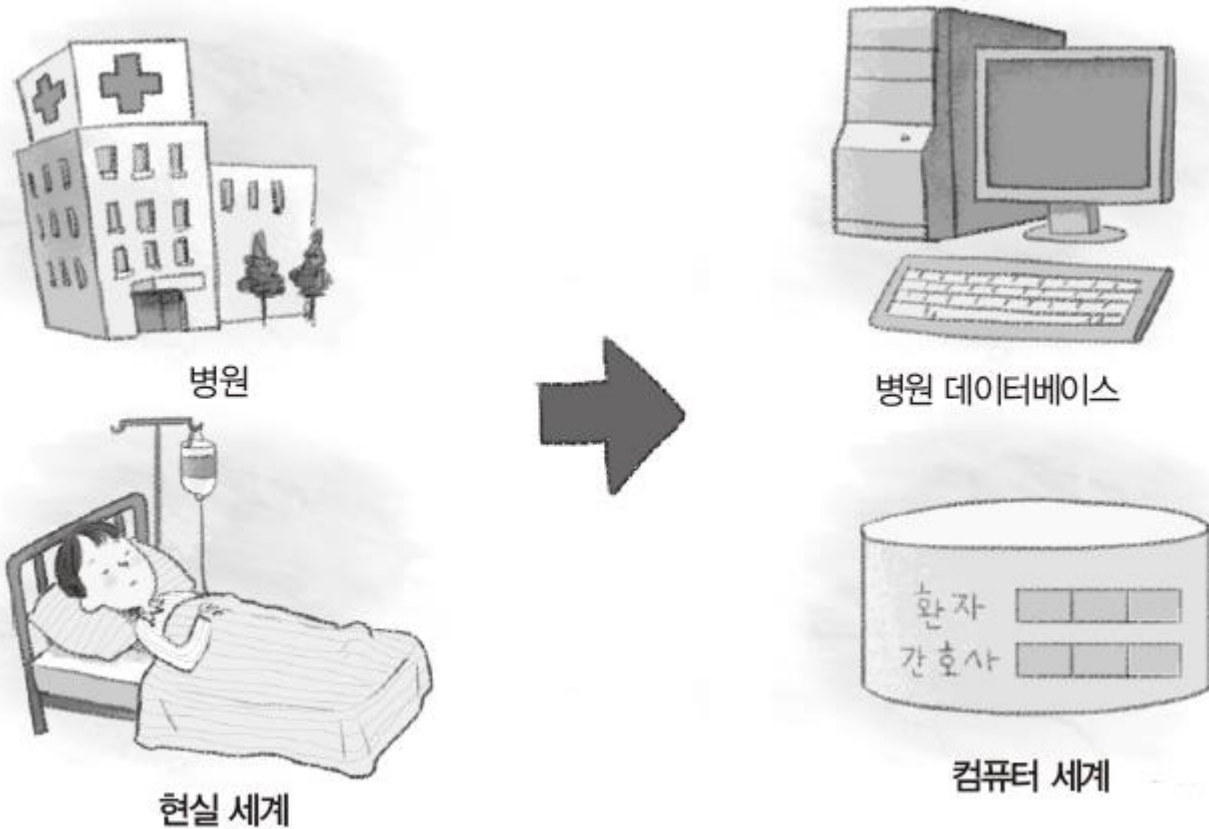
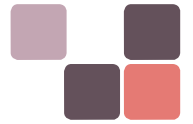
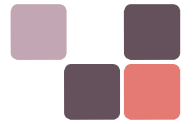


그림 4-1 현실 세계와 컴퓨터 세계



❖ 데이터 모델링(data modeling)

- 현실 세계에 존재하는 데이터를 컴퓨터 세계의 데이터베이스로 옮기는 변환 과정
- 데이터베이스 설계의 핵심 과정

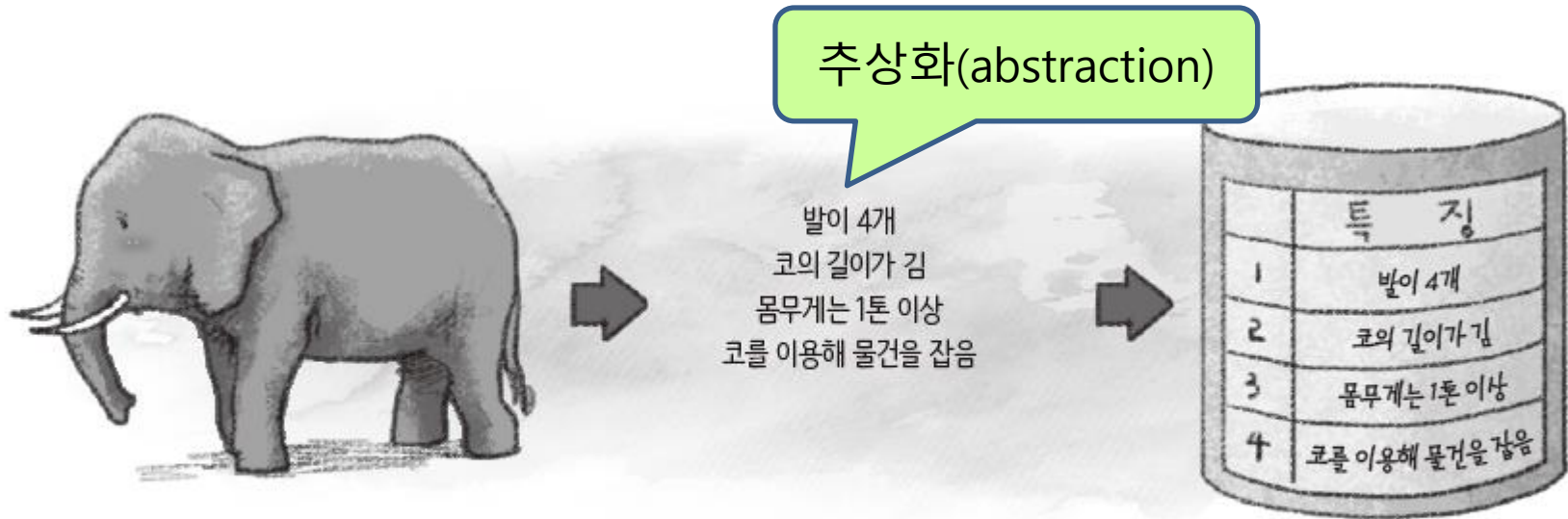
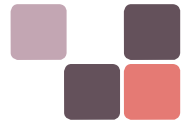


그림 4-2 코끼리의 데이터 모델링 예



❖ 2단계 데이터 모델링

- 개념적 데이터 모델링(conceptual modeling)
 - 현실 세계의 중요 데이터를 추출하여 개념 세계로 옮기는 작업
- 논리적 데이터 모델링(logical modeling)
 - 개념 세계의 데이터를 데이터베이스에 저장하는 구조로 표현하는 작업

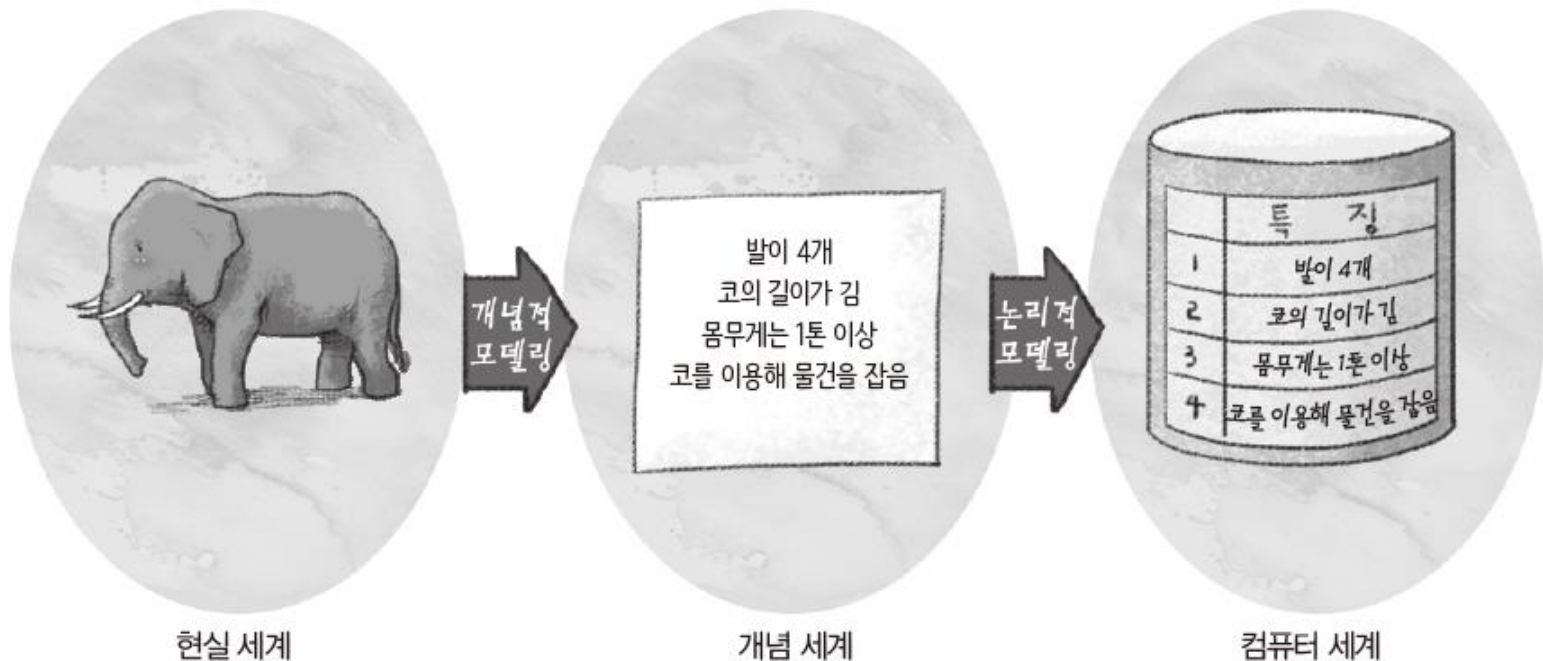
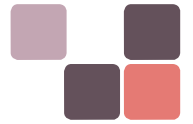


그림 4-3 코끼리의 2단계 데이터 모델링 예



❖ 데이터 모델(data model)

- 데이터 모델링의 결과물을 표현하는 도구
- 개념적 데이터 모델
 - 사람의 머리로 이해할 수 있도록 현실 세계를 개념적 모델링하여 데이터베이스의 개념적 구조로 표현하는 도구
 - 예) 개체-관계 모델
- 논리적 데이터 모델
 - 개념적 구조를 논리적 모델링하여 데이터베이스의 논리적 구조로 표현하는 도구
 - 예) 관계 데이터 모델

01 데이터 모델링과 데이터 모델의 개념

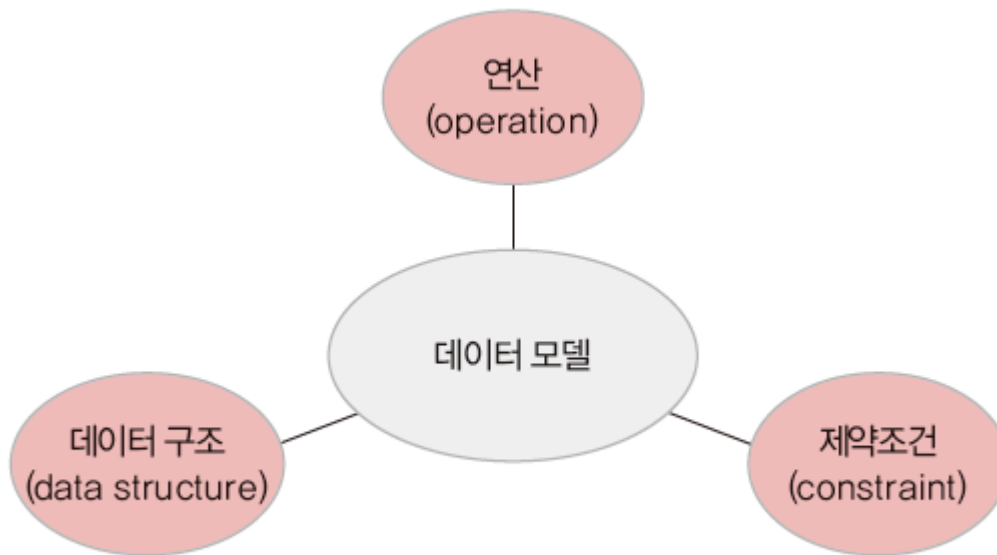
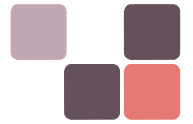
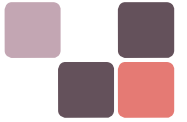


그림 4-4 데이터 모델의 구성



❖ 개체-관계 모델(E-R model; Entity-Relationship model)

- 피터 첸(Peter Chen)이 제안한 개념적 데이터 모델
- 개체와 개체 간의 관계를 이용해 현실 세계를 개념적 구조로 표현
- 핵심 요소 : 개체, 속성, 관계

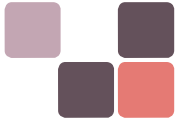
❖ 개체-관계 다이어그램(E-R diagram)

- E-R 다이어그램
- 개체 - 관계 모델을 이용해 현실 세계를 개념적으로 모델링한 결과물을 그림으로 표현한 것



❖ 개체(entity)

- 현실 세계에서 조직을 운영하는 데 꼭 필요한 사람이나 사물과 같이 구별되는 모든 것
- 저장할 가치가 있는 중요 데이터를 가지고 있는 사람이나 사물, 개념, 사건 등
- 다른 개체와 구별되는 이름을 가지고 있고, 각 개체만의 고유한 특성이나 상태, 즉 속성을 하나 이상 가지고 있음
- 예) 서점에 필요한 개체 : 고객, 책
- 예) 학교에 필요한 개체 : 학과, 과목
- 파일 구조의 레코드(record)와 대응됨



❖ 개체

- E-R 다이어그램에서 사각형으로 표현하고 사각형 안에 이름을 표기

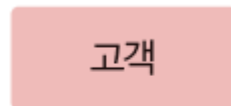


그림 4-6 개체의 E-R 다이어그램 표현 예 : 고객 개체



❖ 속성(attribute)

- 개체나 관계가 가지고 있는 고유의 특성
- 의미 있는 데이터의 가장 작은 논리적 단위
- 파일 구조의 필드(field)와 대응됨
- E-R 다이어그램에서 타원으로 표현하고 타원 안에 이름을 표기

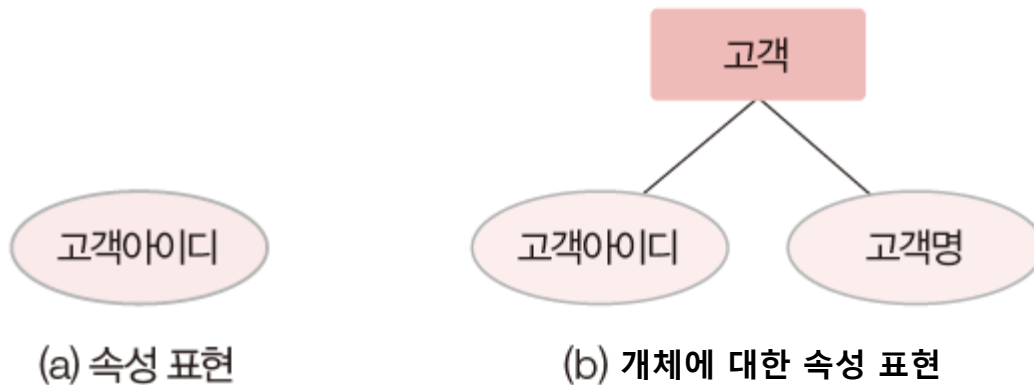
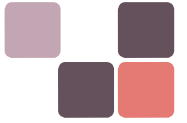


그림 4-7 속성의 E-R 다이어그램 표현 예



❖ 개체 타입(entity type)

- 개체를 고유의 이름과 속성들로 정의한 것
- 파일 구조의 레코드 타입(record type)에 대응됨

❖ 개체 인스턴스(entity instance)

- 개체를 구성하고 있는 속성이 실제 값을 가짐으로써 실체화된 개체
- 개체 어커런스(entity occurrence)라고도 함
- 파일 구조의 레코드 인스턴스(record instance)에 대응됨

❖ 개체 집합(entity set)

- 특정 개체 타입에 대한 개체 인스턴스들을 모아놓은 것

02 개체-관계 모델

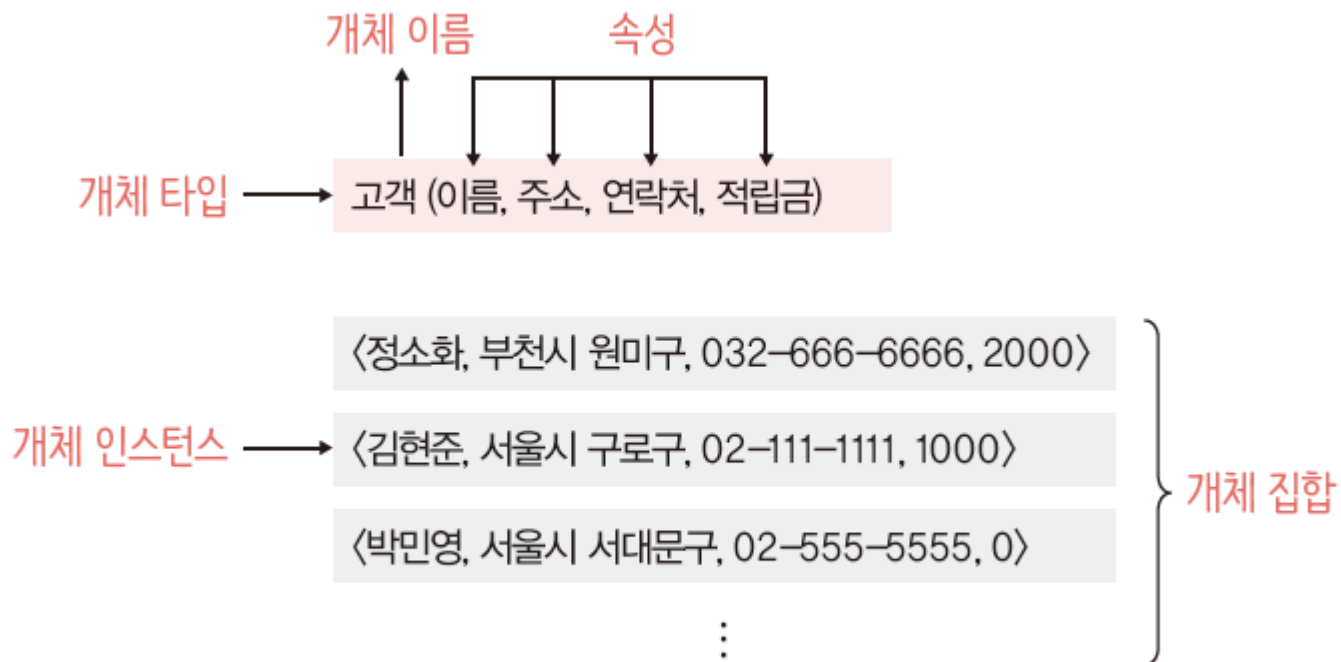
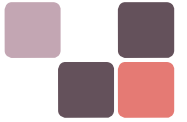


그림 4-5 개체 타입과 개체 인스턴스의 예 : 고객 개체 타입과 고객 개체 인스턴스



❖ 속성의 분류

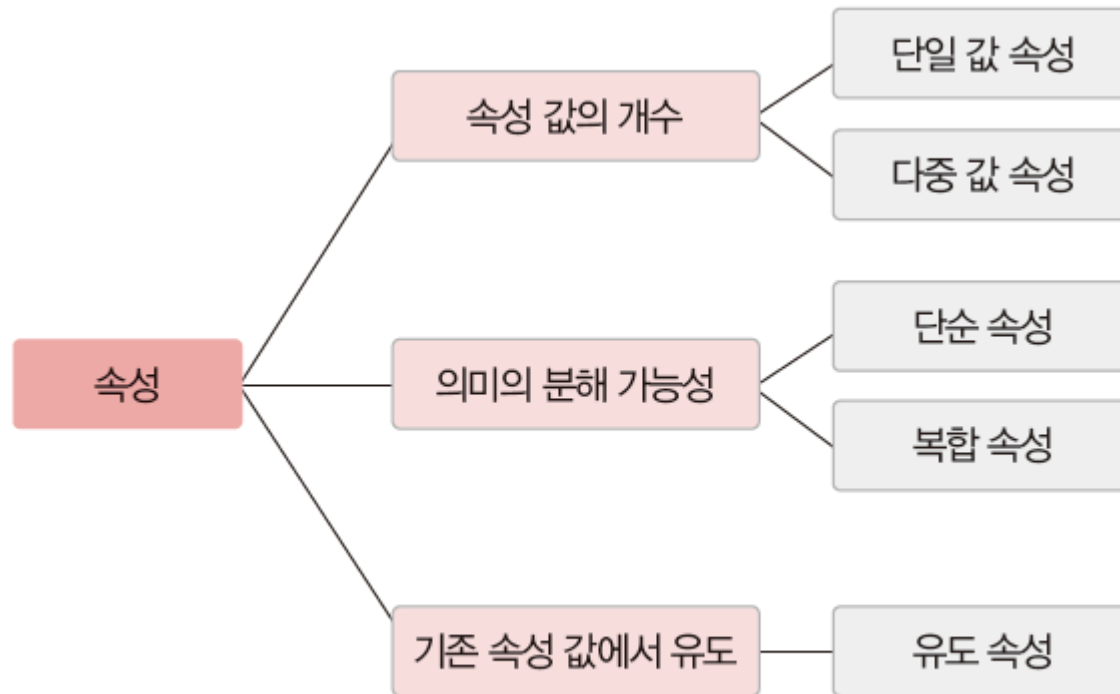


그림 4-8 속성의 분류



❖ 단일 값 속성과 다중 값 속성

- 단일 값 속성(single-valued attribute)
 - 값을 하나만 가질 수 있는 속성
 - 예) 고객 개체의 이름, 적립금 속성
- 다중 값 속성(multi-valued attribute)
 - 값을 여러 개 가질 수 있는 속성
 - 예) 고객 개체의 연락처 속성
 - 예) 책 개체의 저자 속성
 - E-R 다이어그램에서 이중 타원으로 표현

02 개체-관계 모델



❖ 단일 값 속성과 다중 값 속성

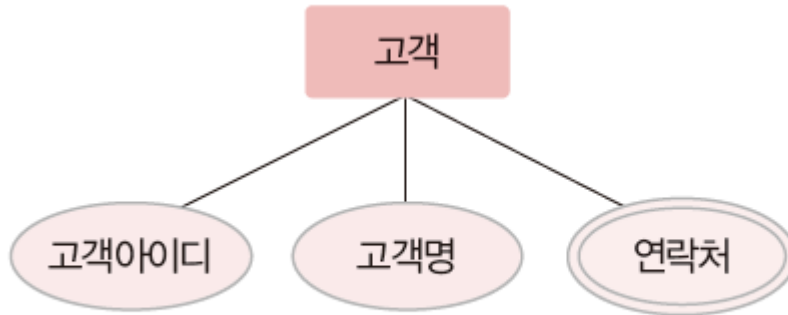
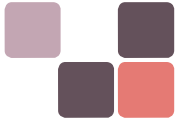


그림 4-9 다중 값 속성의 E-R 다이어그램 표현 예 : 연락처 속성



❖ 단순 속성과 복합 속성

- 단순 속성(simple attribute)
 - 의미를 더는 분해할 수 없는 속성
 - 예) 고객 개체의 적립금 속성
 - 예) 책 개체의 이름, ISBN, 가격 속성
- 복합 속성(composite attribute)
 - 의미를 분해할 수 있는 속성
 - 예) 고객 개체의 주소 속성
 - 도, 시, 동, 우편번호 등으로 의미를 세분화할 수 있음
 - 예) 고객 개체의 생년월일 속성
 - 연, 월, 일로 의미를 세분화할 수 있음



❖ 단순 속성과 복합 속성

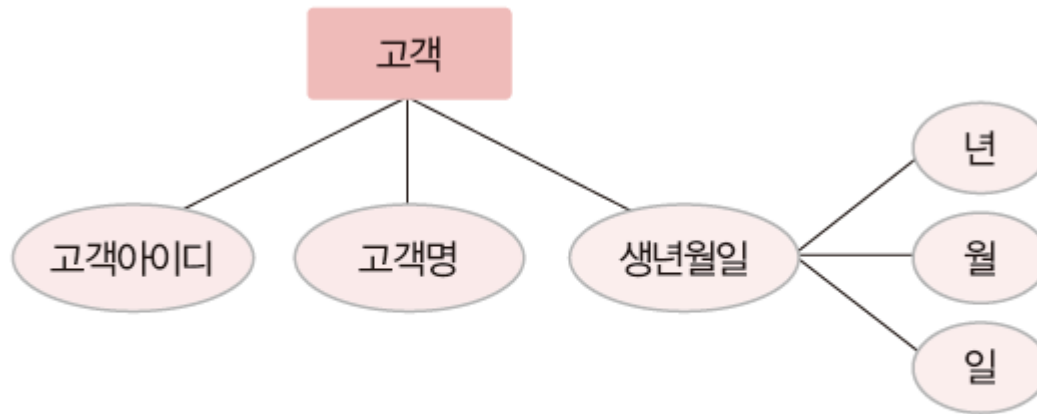


그림 4-10 복합 속성의 E-R 다이어그램 표현 예 : 생년월일 속성



❖ 유도 속성(derived attribute)

- 기존의 다른 속성의 값에서 유도되어 결정되는 속성
- 값이 별도로 저장되지 않음
- 예) 책 개체의 가격과 할인을 속성으로 계산되는 판매가격 속성
- 예) 고객 개체의 출생연도 속성으로 계산되는 나이 속성
- E-R 다이어그램에서 점선 타원으로 표현

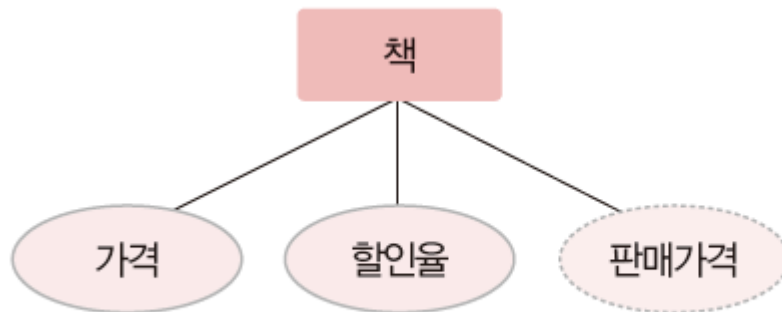
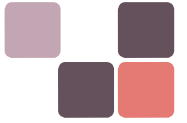


그림 4-11 유도 속성의 E-R 다이어그램 표현 예 : 판매가격 속성



❖ 널 속성(null attribute)

- 널 값이 허용되는 속성

❖ 널(null) 값

- 아직 결정되지 않았거나 모르는 값 또는 존재하지 않는 값
- 공백이나 0과는 의미가 다름
- 예) 등급 속성이 널 값 → 등급이 아직 결정되지 않았음을 의미



❖ 키 속성(key attribute)

- 각 개체 인스턴스를 식별하는 데 사용되는 속성
- 모든 개체 인스턴스의 키 속성 값이 다름
- 둘 이상의 속성들로 구성되기도 함
- 예) 고객 개체의 고객아이디 속성
- E-R 다이어그램에서 밑줄로 표현

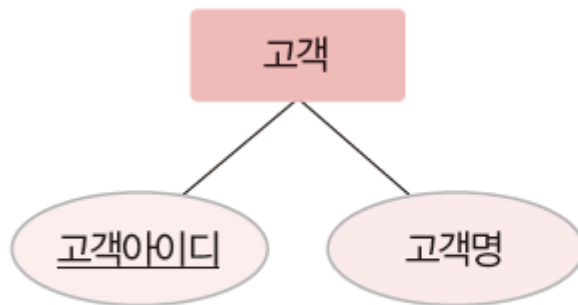


그림 4-12 키 속성의 E-R 다이어그램 표현 예 : 고객아이디 속성



❖ 관계(relationship)

- 개체와 개체가 맺고 있는 의미 있는 연관성
- 개체 집합들 사이의 대응 관계, 즉 매핑(mapping)을 의미
- 예) 고객 개체와 책 개체 간의 구매 관계
 - “고객은 책을 구매한다”
- E-R 다이어그램에서 마름모로 표현

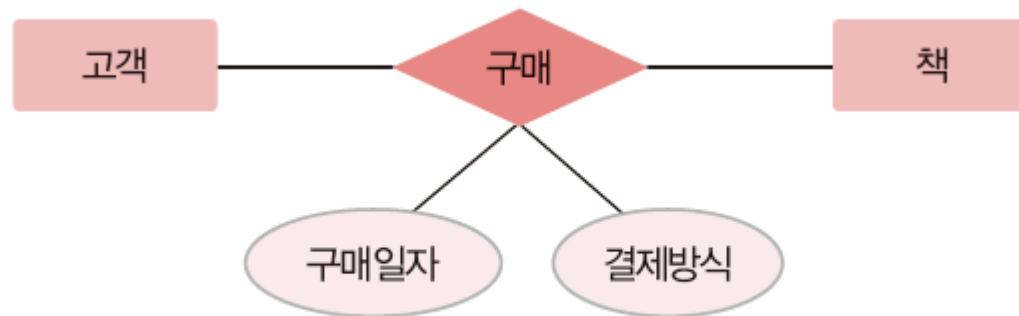


그림 4-13 관계의 E-R 다이어그램 표현 예 : 구매 관계



❖ 관계의 유형 : 관계에 참여하는 개체 타입의 수 기준

- 이항 관계 : 개체 타입 두 개가 맺는 관계
- 삼항 관계 : 개체 타입 세 개가 맺는 관계
- 순환 관계 : 개체 타입 하나가 자기 자신과 맺는 관계



❖ 관계의 유형 : 매핑 카디널리티 기준

- 일대일(1:1) 관계
- 일대다(1:n)관계
- 다대다(n:m) 관계

❖ 매핑 카디널리티(mapping cardinality)

- 관계를 맺는 두 개체 집합에서, 각 개체 인스턴스가 연관성을 맺고 있는 상대 개체 집합의 인스턴스 개수



❖ 일대일(1:1) 관계

- 개체 A의 각 개체 인스턴스가 개체 B의 개체 인스턴스 **하나**와 관계를 맺을 수 있고, 개체 B의 각 개체 인스턴스도 개체 A의 개체 인스턴스 **하나**와 관계를 맺을 수 있음

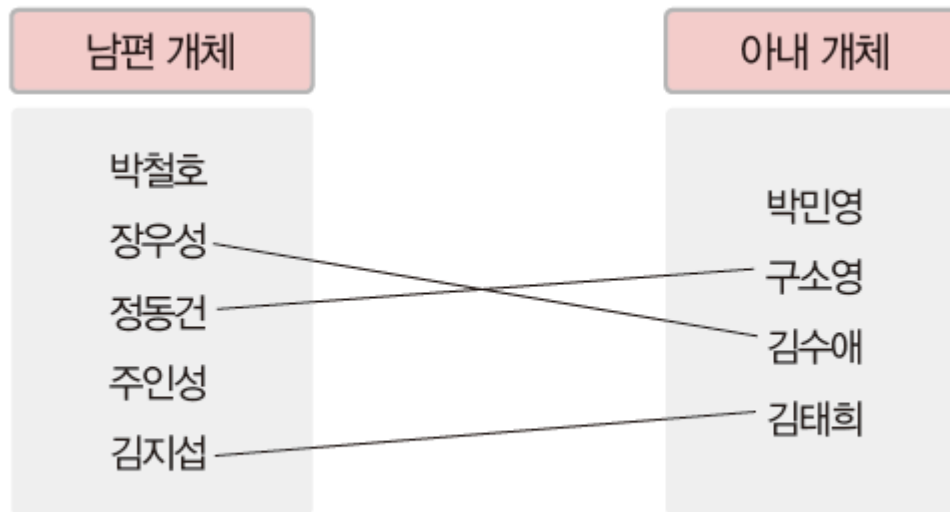
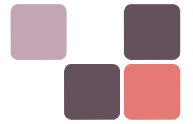


그림 4-14 일대일 관계의 예 : 남편과 아내 개체의 혼인 관계



❖ 일대다(1:n) 관계

- 개체 A의 각 개체 인스턴스가 개체 B의 개체 인스턴스 **여러 개**와 관계를 맺을 수 있지만, 개체 B의 각 개체 인스턴스는 개체 A의 개체 인스턴스 **하나**와 관계를 맺을 수 있음

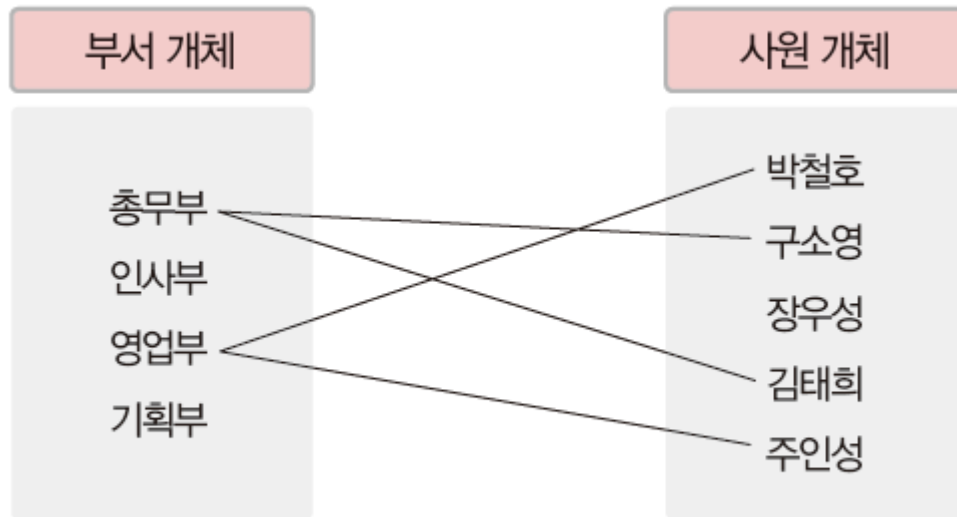
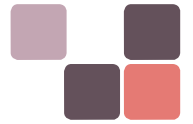


그림 4-15 일대다 관계의 예 : 부서와 직원 개체의 소속 관계



❖ 다대다(n:m) 관계

- 개체 A의 각 개체 인스턴스가 개체 B의 개체 인스턴스 **여러 개**와 관계를 맺을 수 있고, 개체 B의 각 개체 인스턴스도 개체 A의 개체 인스턴스 **여러 개**와 관계를 맺을 수 있음

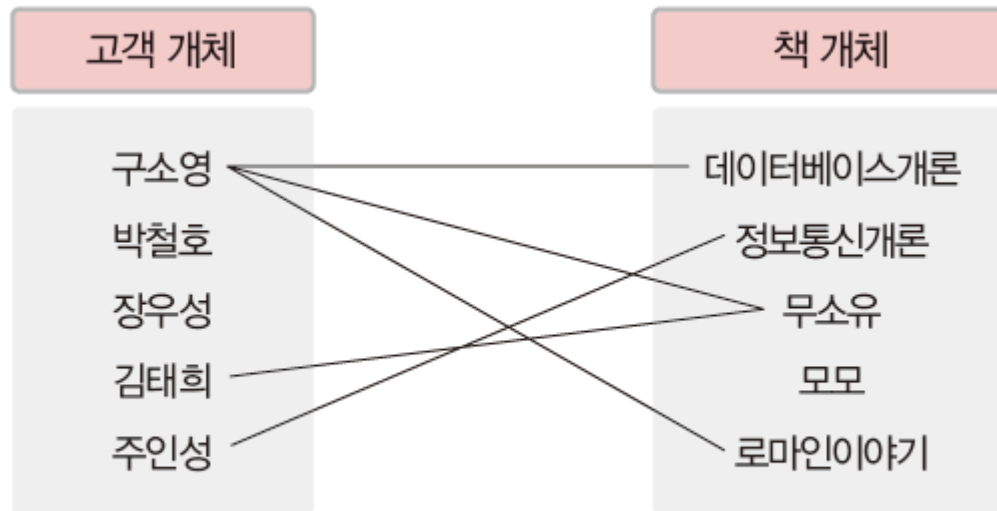
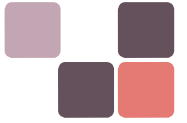


그림 4-16 다대다 관계의 예 : 고객과 책 개체의 구매 관계



❖ 관계의 참여 특성

- 필수적 참여(전체 참여)
 - 모든 개체 인스턴스가 관계에 반드시 참여해야 되는 것을 의미
 - 예) 고객 개체가 책 개체와의 구매 관계에 필수적으로 참여
 - 모든 고객은 책을 반드시 구매해야 함
 - E-R 다이어그램에서 이중선으로 표현
- 선택적 참여(부분 참여)
 - 개체 인스턴스 중 일부만 관계에 참여해도 되는 것을 의미
 - 예) 책 개체가 고객 개체와의 구매 관계에 선택적으로 참여
 - 고객이 구매하지 않은 책이 존재할 수 있음



❖ 관계의 참여 특성



그림 4-17 필수적 참여 관계의 E-R 다이어그램 표현 예 : 고객 개체의 필수적 참여 관계



❖ 관계의 종속성

- 약한 개체(weak entity)
 - 다른 개체의 존재 여부에 의존적인 개체
- 강한 개체(strong entity)
 - 다른 개체의 존재 여부를 결정하는 개체
- 특징
 - 강한 개체와 약한 개체는 일반적으로 일대다의 관계를 가지고, 약한 개체는 강한 개체와의 관계에 필수적으로 참여함
 - 약한 개체는 강한 개체의 키를 포함하여 키를 구성함



❖ 관계의 종속성

- E-R 다이어그램에서 약한 개체는 이중 사각형으로 표현하고, 약한 개체가 강한 개체와 맺는 관계는 이중 마름모로 표현
- 예) 직원 개체와 부양가족 개체 사이의 부양 관계
 - 직원 개체는 강한 개체, 부양가족 개체는 약한 개체



그림 4-18 관계 종속성의 E-R 다이어그램 표현 예 : 약한 개체인 부양가족 개체



❖ E-R 다이어그램

- 사각형 : 개체를 표현
- 마름모 : 관계를 표현
- 타원 : 속성을 표현
- 링크(연결선) : 각 요소를 연결
- 레이블 : 일대일, 일대다, 다대다 관계를 표기

02 개체-관계 모델

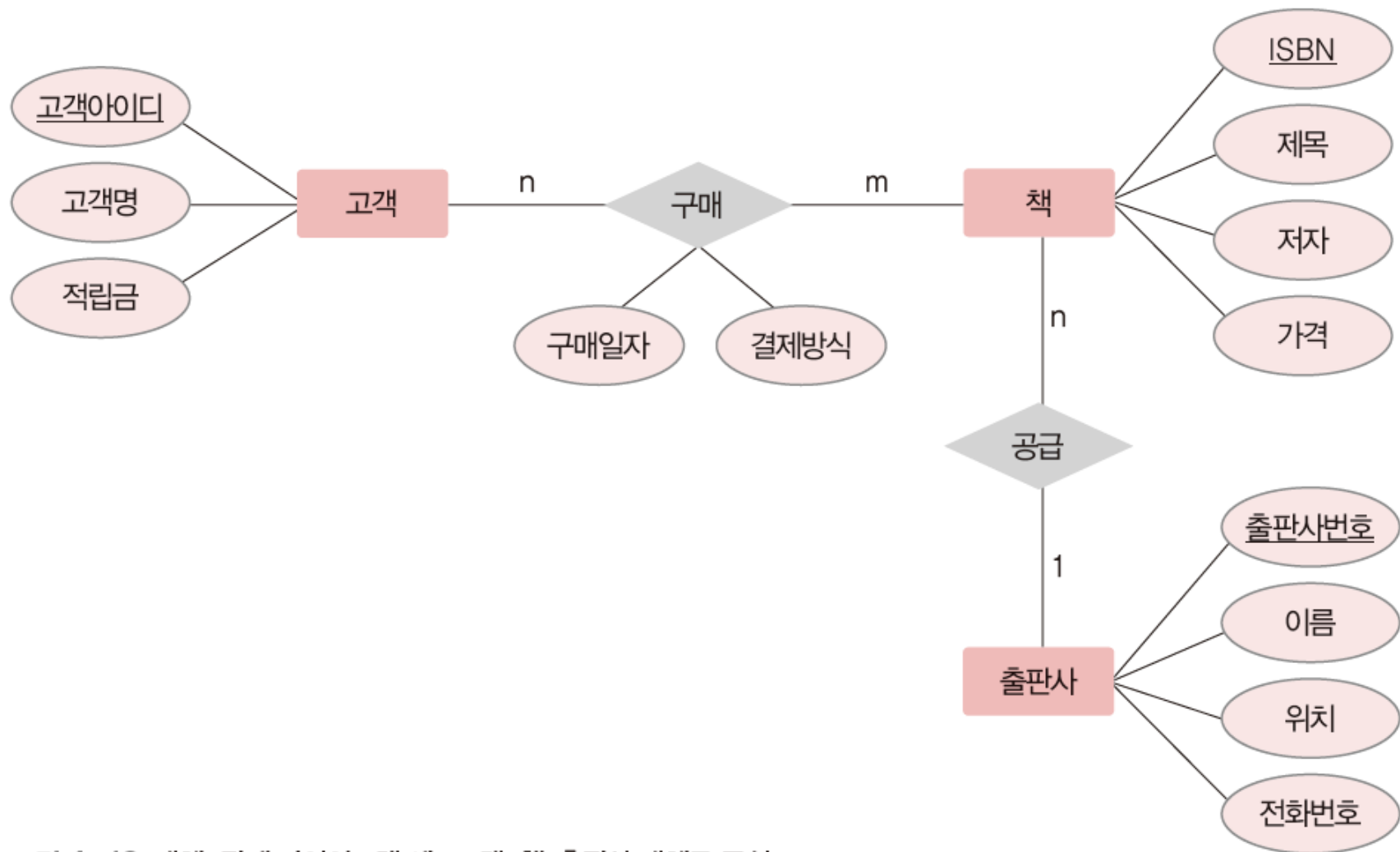
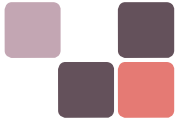


그림 4-19 개체-관계 다이어그램 예 : 고객, 책, 출판사 개체로 구성



❖ 논리적 데이터 모델의 개념

- E-R 다이어그램으로 표현된 개념적 구조를 데이터베이스에 저장할 형태로 표현한 논리적 구조
 - 데이터베이스의 논리적 구조 = 데이터베이스 스키마(schema)
- 사용자가 생각하는 데이터베이스의 모습 또는 구조
- 관계 데이터 모델, 계층 데이터 모델, 네트워크 데이터 모델 등이 있음



❖ 관계 데이터 모델

- 일반적으로 많이 사용되는 논리적 데이터 모델
- 데이터베이스의 논리적 구조가 2차원 테이블 형태임



❖ 계층 데이터 모델(hierarchical data model)

- 데이터베이스의 논리적 구조가 트리(tree) 형태임
- 루트 역할을 하는 개체가 존재하고 사이클이 존재하지 않음
- 개체 간에 상하 관계가 성립
 - 부모 개체 / 자식 개체
 - 부모와 자식 개체는 일대다(1:n) 관계만 허용됨
- 두 개체 사이에 하나의 관계만 정의할 수 있음
- 다대다(n:m) 관계를 직접 표현할 수 없음
- 개념적 구조를 모델링하기 어려워 구조가 복잡해질 수 있음
- 데이터의 삽입·삭제·수정·검색이 쉽지 않음

03 논리적 데이터 모델



❖ 계층 데이터 모델의 예

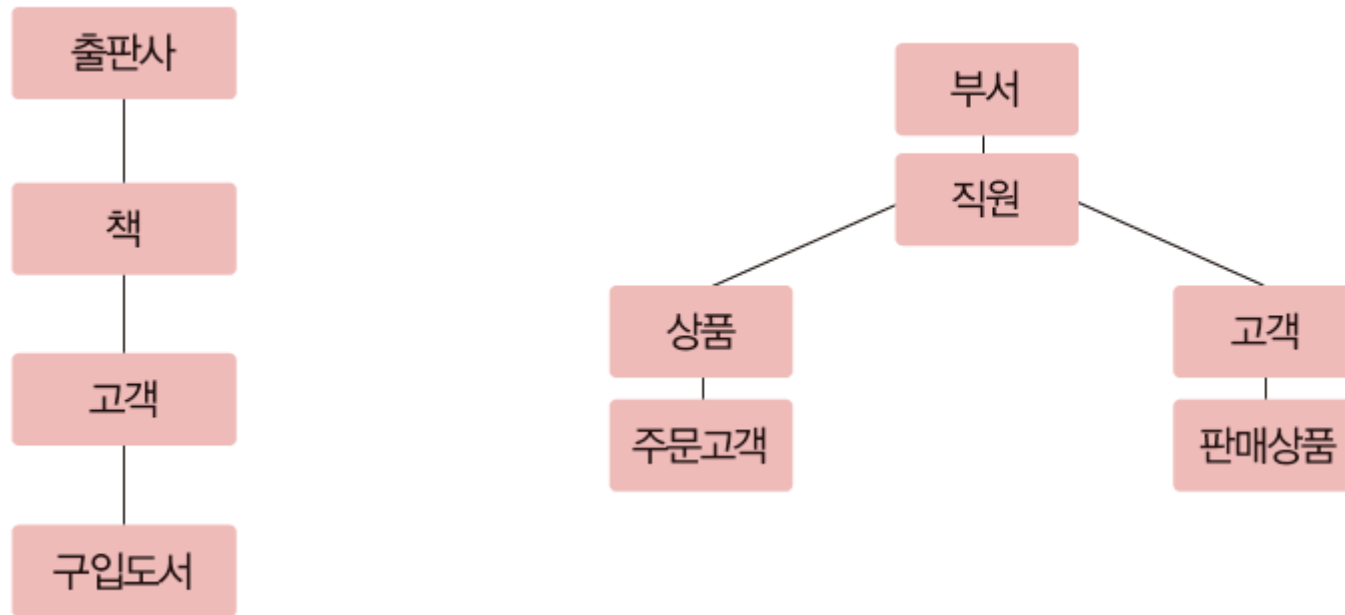


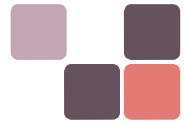
그림 4-20 계층 데이터 모델의 예



❖ 네트워크 데이터 모델(network data model)

- 데이터베이스의 논리적 구조가 네트워크, 즉 그래프 형태임
- 개체 간에는 일대다(1:n) 관계만 허용됨
 - 오너(owner) / 멤버(member)
- 두 개체 사이에 여러 관계를 정의할 수 있어 이름으로 구별함
- 다대다(n:m) 관계를 직접 표현할 수 없음
- 구조가 복잡하고 데이터의 삽입·삭제·수정·검색이 쉽지 않음

03 논리적 데이터 모델



❖ 네트워크 데이터 모델의 예

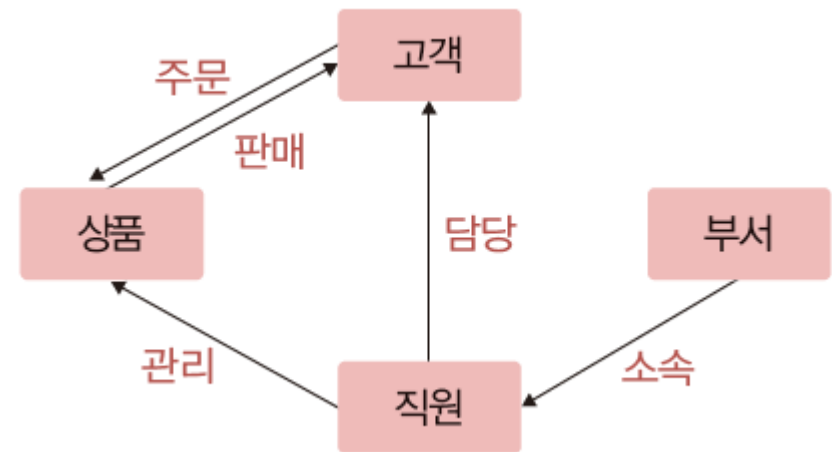
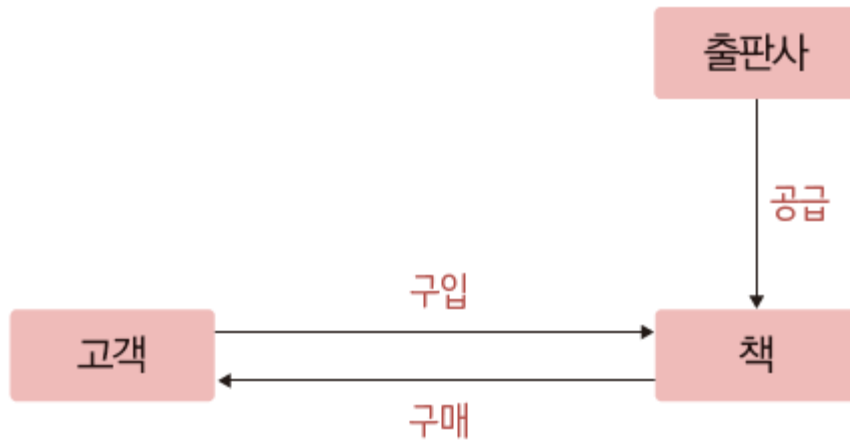
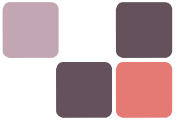


그림 4-21 네트워크 데이터 모델의 예



Thank You