

# Database

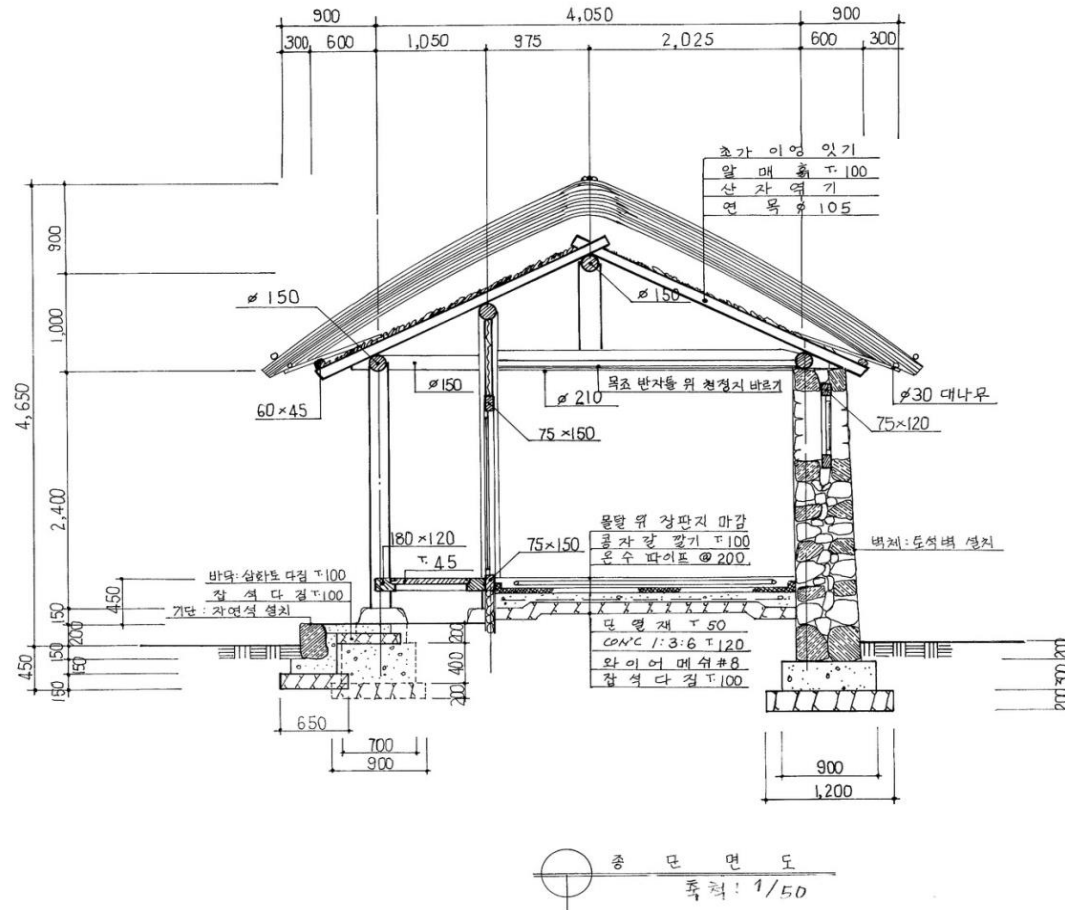
## 데이터 모델링

# 01. 데이터 모델링의 개념

---

- 데이터베이스 생명주기
- 데이터 모델링 과정

# 01. 데이터 모델링의 개념



건물 설계  
(소프트웨어 설계)

지반 설계  
(데이터베이스 설계)

그림 6-1 데이터 모델링의 중요성

# 01. 데이터 모델링의 개념

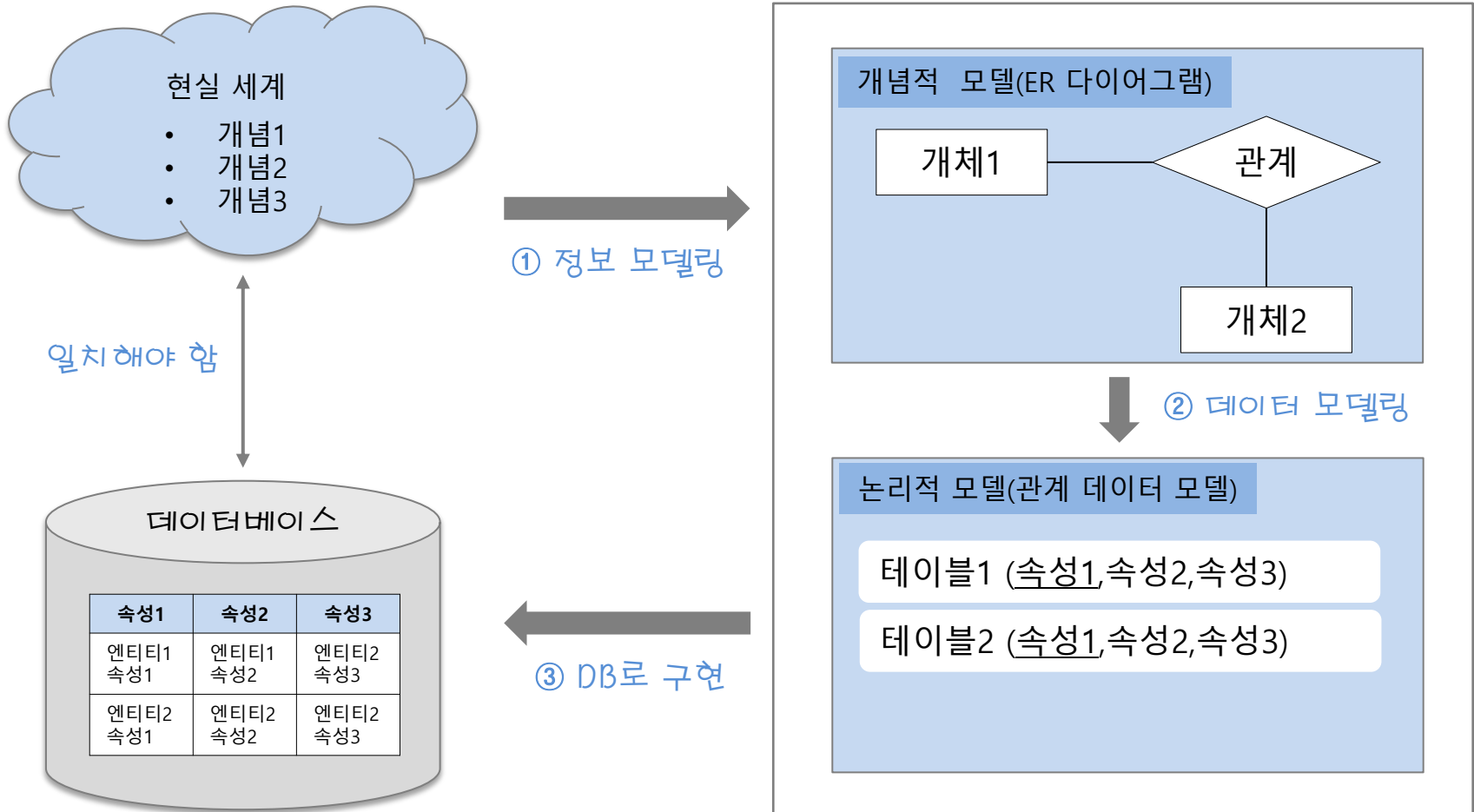


그림 6-2 데이터 모델링의 개념

# 1.1 데이터베이스 생명주기

- 데이터베이스 생명주기(database life cycle)란?  
데이터베이스의 생성과 운영에 관련된 특징

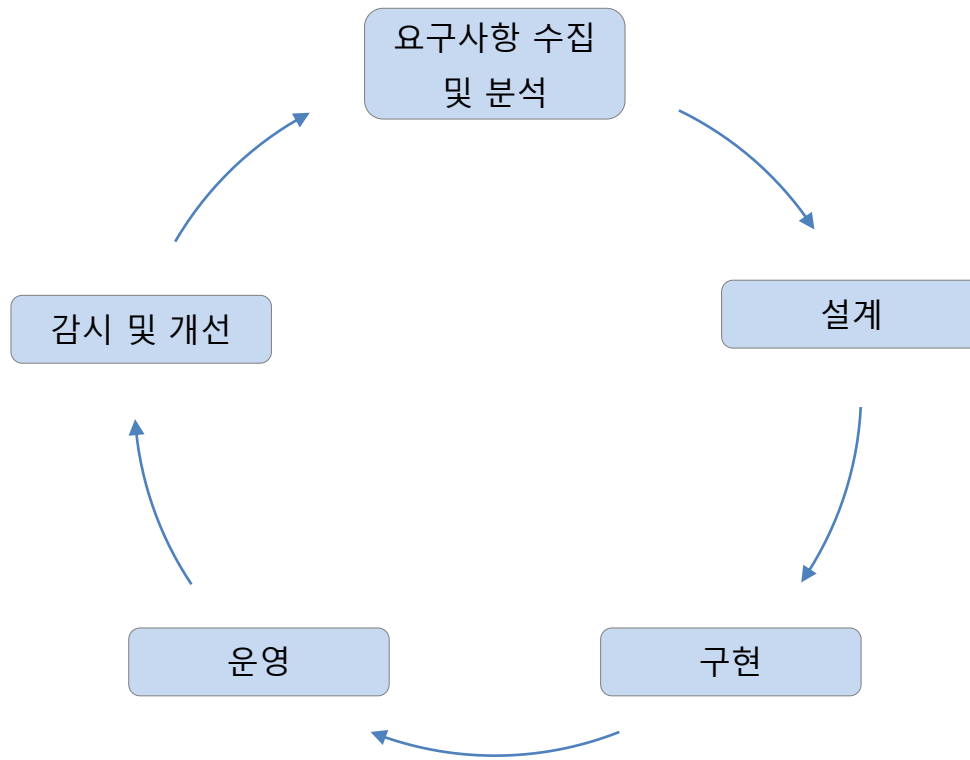


그림 6-3 데이터베이스 생명주기

# 1.1 데이터베이스 생명주기

## ① 요구사항 수집 및 분석

사용자들의 요구사항을 듣고 분석하여 데이터베이스 구축의 범위를 정하는 단계

## ② 설계

분석된 요구사항을 기초로 주요 개념과 업무 프로세스 등을 식별하고(개념적 설계), 사용하는 DBMS의 종류에 맞게 변환(논리적 설계)한 후, 데이터베이스 스키마를 도출(물리적 설계)함.

## ③ 구현

설계 단계에서 생성한 스키마를 실제 DBMS에 적용하여 테이블 및 관련 객체(뷰, 인덱스 등)를 만듦.

## ④ 운영

구현된 데이터베이스를 기반으로 소프트웨어를 구축하여 서비스를 제공함.

## ⑤ 감시 및 개선

데이터베이스 운영에 따른 시스템의 문제를 관찰하고 데이터베이스 자체의 문제점을 파악하여 개선함.

## 1.2 데이터 모델링 과정

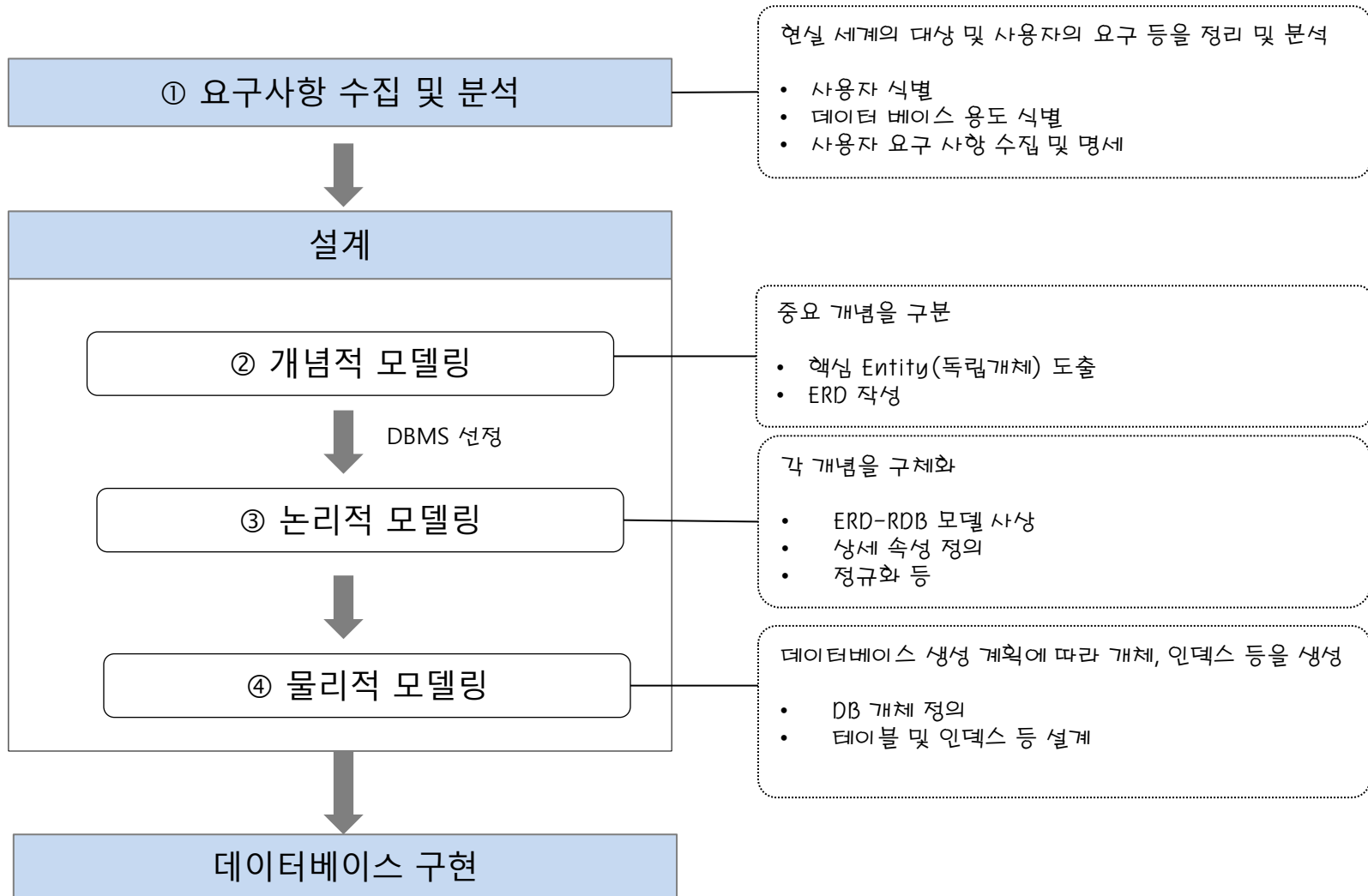


그림 6-4 데이터 모델링 과정

## 1.2.1 요구사항 수집 및 분석

---

### ■ 요구사항 수집 방법

1. 실제 문서를 수집하고 분석함.
2. 담당자와의 인터뷰나 설문조사를 통해 요구사항을 직접 수렴함.
3. 비슷한 업무를 처리하는 기존의 데이터베이스를 분석함.
4. 각 업무와 연관된 모든 부분을 살펴봄.



## 1.2.2 개념적 모델링

- 개념적 모델링(conceptual modeling) : 요구사항을 수집하고 분석한 결과를 토대로 업무의 핵심적인 개념을 구분하고 전체적인 뼈대를 만드는 과정
- 개체(entity)를 추출하고 각 개체들 간의 관계를 정의하여 ER 다이어그램(ERD, Entity Relationship Diagram)을 만드는 과정까지를 말함.

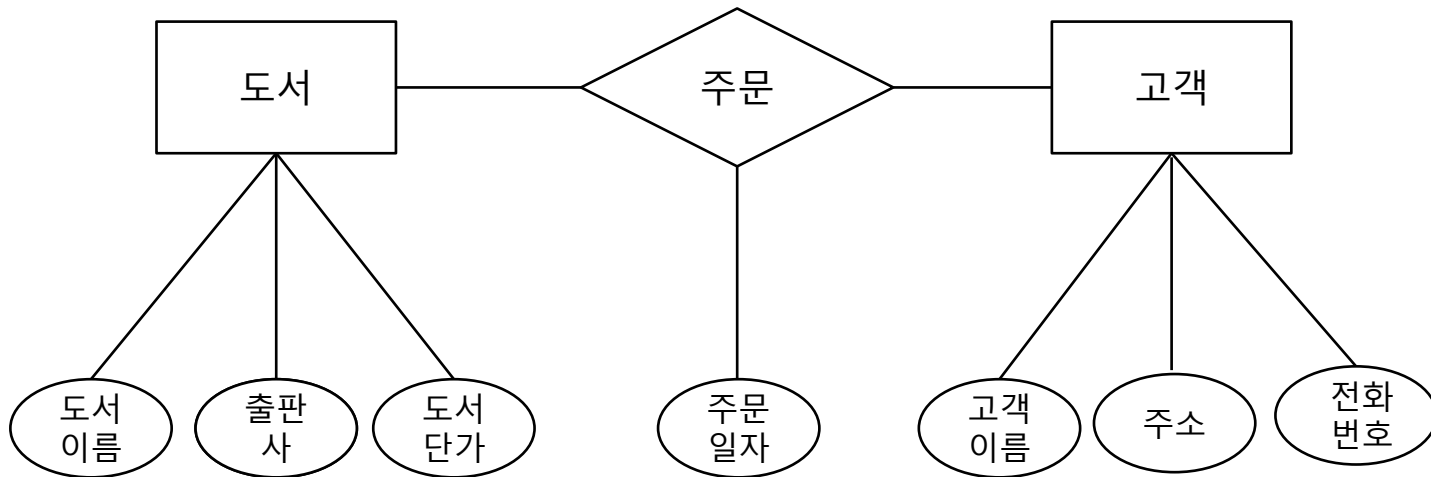


그림 6-5 개념적 모델링의 예

## 1.2.3 논리적 모델링

### ■ 논리적 모델링(logical modeling)이란?

- 개념적 모델링에서 만든 ER 다이어그램을 사용하려는 DBMS에 맞게 사상(매핑, mapping)하여 실제 데이터베이스로 구현하기 위한 모델을 만드는 과정



도서 (도서번호, 도서이름, 출판사이름, 도서단가)

고객 (고객번호, 고객이름, 주소, 전화번호)

주문 (주문번호, 고객번호(FK), 도서번호(FK), 주문일자, 주문금액)

그림 6-6 논리적 모델링의 예

### ■ 논리적 모델링 과정

1. 개념적 모델링에서 추출하지 않았던 상세 속성들을 모두 추출함.
2. 정규화 수행
3. 데이터 표준화 수행

## 1.2.4 물리적 모델링

### ■ 물리적 모델링(physical modeling)이란?

- 작성된 논리적 모델을 실제 컴퓨터의 저장 장치에 저장하기 위한 물리적 구조를 정의하고 구현하는 과정
- DBMS의 특성에 맞게 저장 구조를 정의해야 데이터베이스가 최적의 성능을 낼 수 있음.

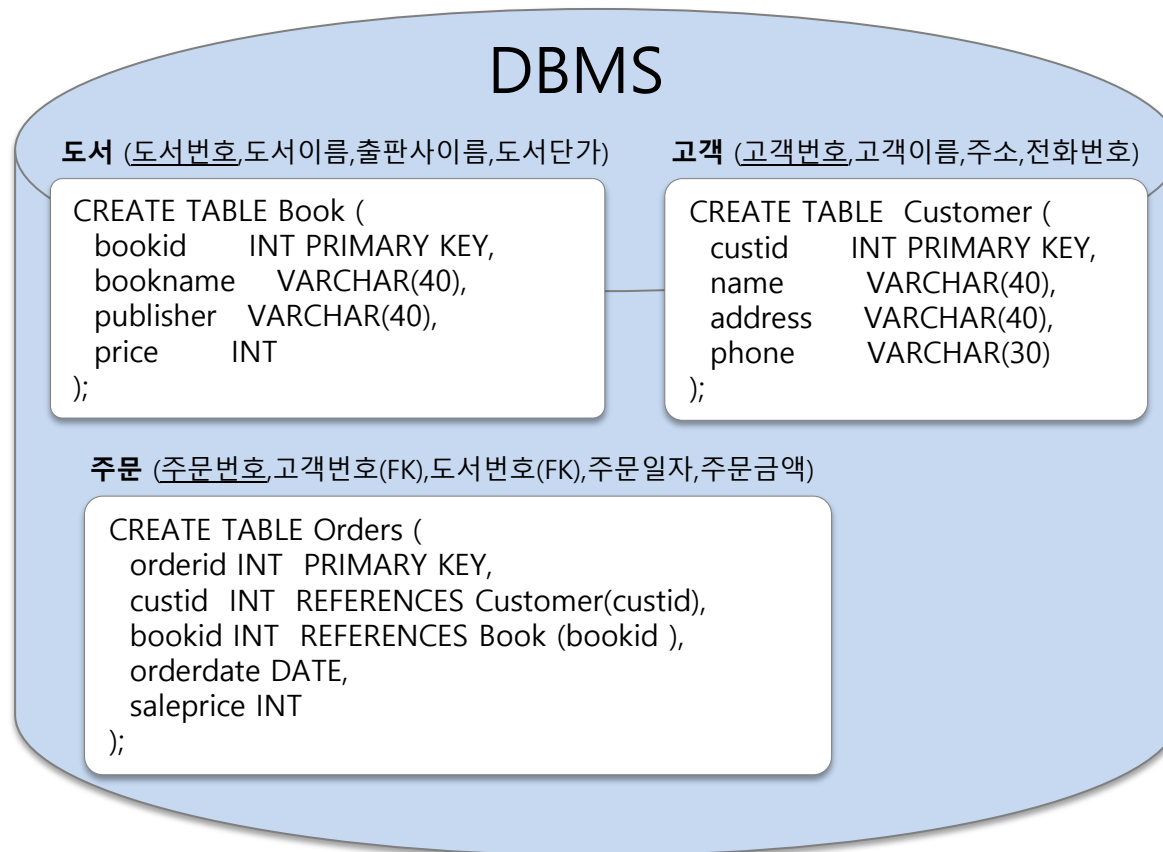


그림 6-7 물리적 모델링의 예

## 1.2.4 물리적 모델링

### ■ 물리적 모델링 시 트랜잭션, 저장 공간 설계 측면에서 고려할 사항

1. 응답시간을 최소화해야 한다.
2. 얼마나 많은 트랜잭션을 동시에 발생시킬 수 있는지 검토해야 한다.
3. 데이터가 저장될 공간을 효율적으로 배치해야 한다.

## 02. ER 모델

---

- 개체와 개체 타입
- 속성
- 관계와 관계 타입
- 약한 개체 타입과 식별자
- IE 표기법

## 02. ER 모델

### ■ ER(Entity Relationship) 모델

- 세상의 사물을 개체(entity)와 개체 간의 관계(relationship)로 표현함.

### ■ 개체

- 독립적인 의미를 지니고 있는 유무형의 사람 또는 사물
- 개체의 특성을 나타내는 속성(attribute)에 의해 식별됨. 개체끼리 서로 관계를 가짐.

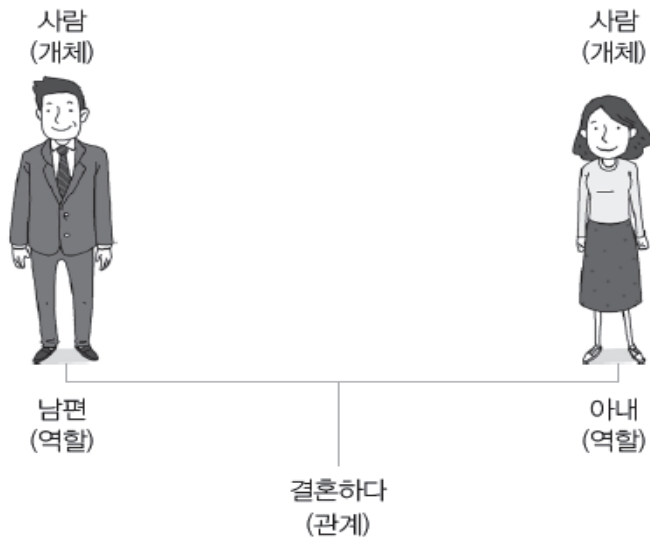


그림 6-8 ER 모델의 기본 개념

## 02. ER 모델

### ■ ER 다이어그램이란?

- ER 모델은 개체와 개체 간의 관계를 표준화된 그림으로 나타냄.

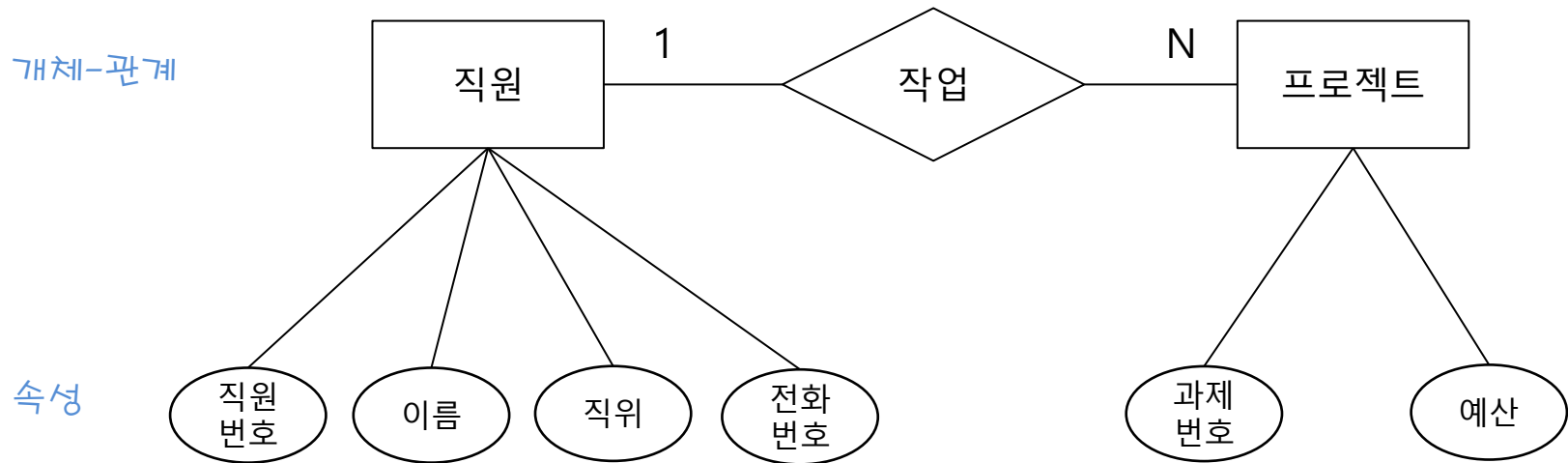


그림 6-9 ER 다이어그램

## 2.1 개체와 개체 타입

### ■ 개체(entity)란?

- 사람, 사물, 장소, 개념, 사건과 같이 유무형의 정보를 가지고 있는 독립적인 실체.
- 데이터베이스에서 주로 다루는 개체는 낱개로 구성된 것, 낱개가 각각 데이터 값을 가지는 것, 데이터 값이 변하는 것 등이 있음.
- 비슷한 속성의 개체 타입(entity type)을 구성하며, 개체 집합(entity set)으로 묶임.

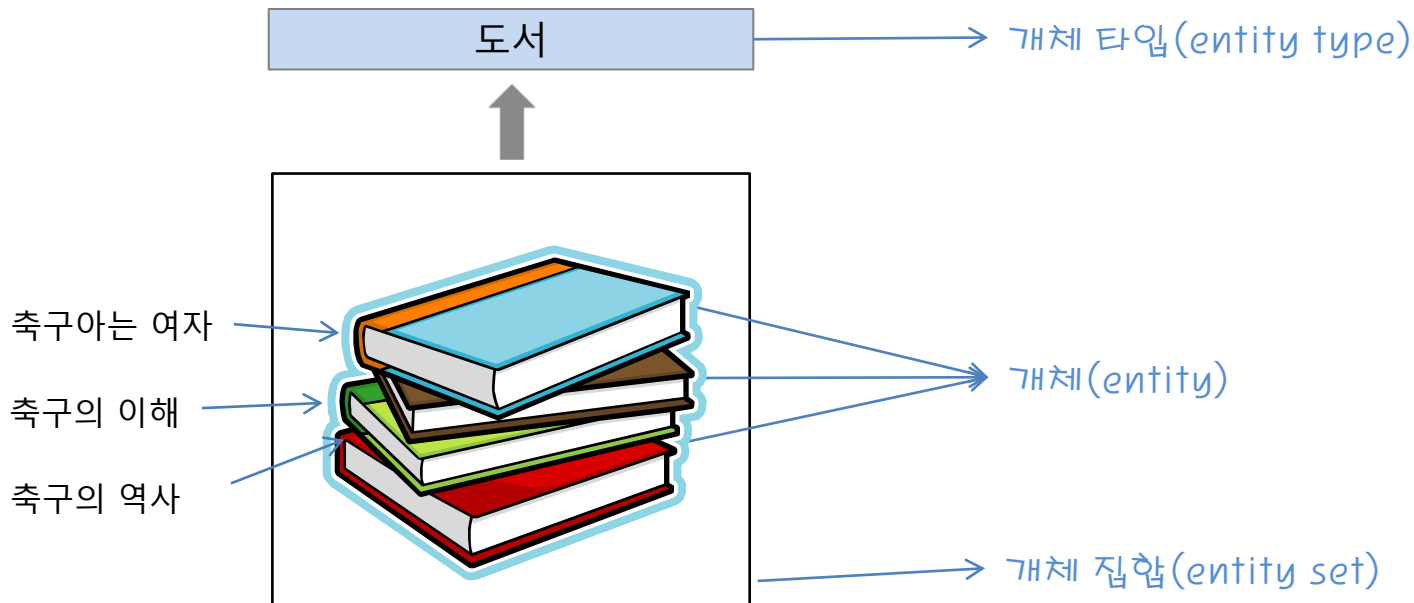



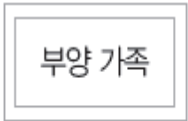
그림 6-10 개체, 개체 타입, 개체 집합



## 2.1.1 개체 타입의 ER 다이어그램 표현

- ER 다이어그램상에서 개체 타입은 직사각형으로 나타냄.

표 6-1 개체 타입의 ER 다이어그램 표현

기호	의미
 직원	강한 개체 타입(보통 개체 타입이라고 하면 강한 개체 타입을 말한다)
 부양 가족	약한 개체 타입

- 개체 타입의 유형
  - 강한 개체(strong entity) : 다른 개체의 도움 없이 독자적으로 존재할 수 있는 개체
  - 약한 개체(weak entity) : 독자적으로는 존재할 수 없고 반드시 상위 개체 타입을 가짐.

## 2.2 속성

### ■ 속성(attribute) : 개체가 가진 성질

개체 타입	속성
도서	도서이름, 출판사, 도서단가

표 6-2 개체 타입과 속성

### ■ 속성의 ER 다이어그램 표현

- 속성은 기본적으로 타원으로 표현. 개체 타입을 나타내는 직사각형과 실선으로 연결됨.
- 속성의 이름은 타원의 중앙에 표기함.
- 속성이 개체를 유일하게 식별할 수 있는 키일 경우 속성 이름에 밑줄을 그음.

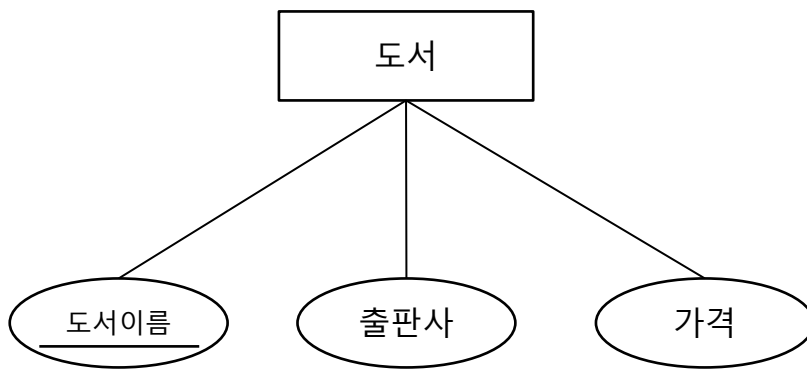


그림 6-11 도서 개체 타입

## 2.2.2 속성의 유형

기호	의미	설명
	속성	<ul style="list-style-type: none"> <li>일반적인 속성을 나타냄</li> <li>속성의 이름은 타원 중앙에 표시</li> </ul>
	키(key) 속성	<ul style="list-style-type: none"> <li>속성이 개체를 유일하게 식별할 수 있는 키일 경우 속성 이름에 밑줄을 그음</li> </ul>
	약한 개체의 식별자	<ul style="list-style-type: none"> <li>약한 개체는 키를 갖지 못하고 대신 식별자를 가짐</li> <li>식별자의 아래에 점선을 그음</li> </ul>
	다중값 속성	<ul style="list-style-type: none"> <li>취미(수영, 자전거)와 같이 여러 개의 값을 갖는 속성</li> <li>이중 타원으로 표현</li> </ul>
	유도 속성	<ul style="list-style-type: none"> <li>나이와 같이 출생년도로 유도가 가능한 속성</li> <li>점선 타원으로 표현</li> </ul>
	복합 속성	<ul style="list-style-type: none"> <li>주소(시, 동, 번지)와 같이 여러 속성으로 구성된 속성</li> <li>큰 타원 아래 작은 타원으로 연결</li> </ul>

그림 6-11  
속성의  
ER 다이어그램 표현

## 2.3 관계와 관계 타입

- 관계(relationship) : 개체 사이의 연관성을 나타내는 개념.
- 관계 타입(relationship type) : 개체 타입과 개체 타입 간의 연결 가능한 관계를 정의한 것이며, 관계 집합(relationship set)은 관계로 연결된 집합을 의미함.

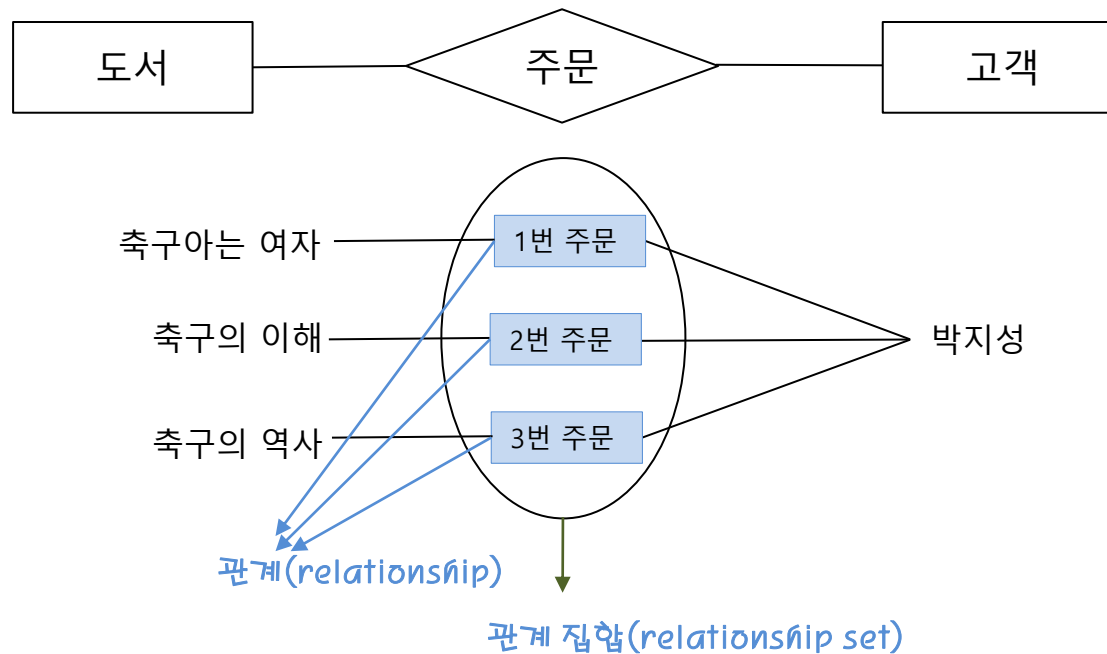



그림 6-15 관계, 관계 타입, 관계 집합

## 2.3.1 관계 타입의 ER 다이어그램 표현

표 6-4 관계 타입의 ER 다이어그램 표현

기호	의미
	관계 타입

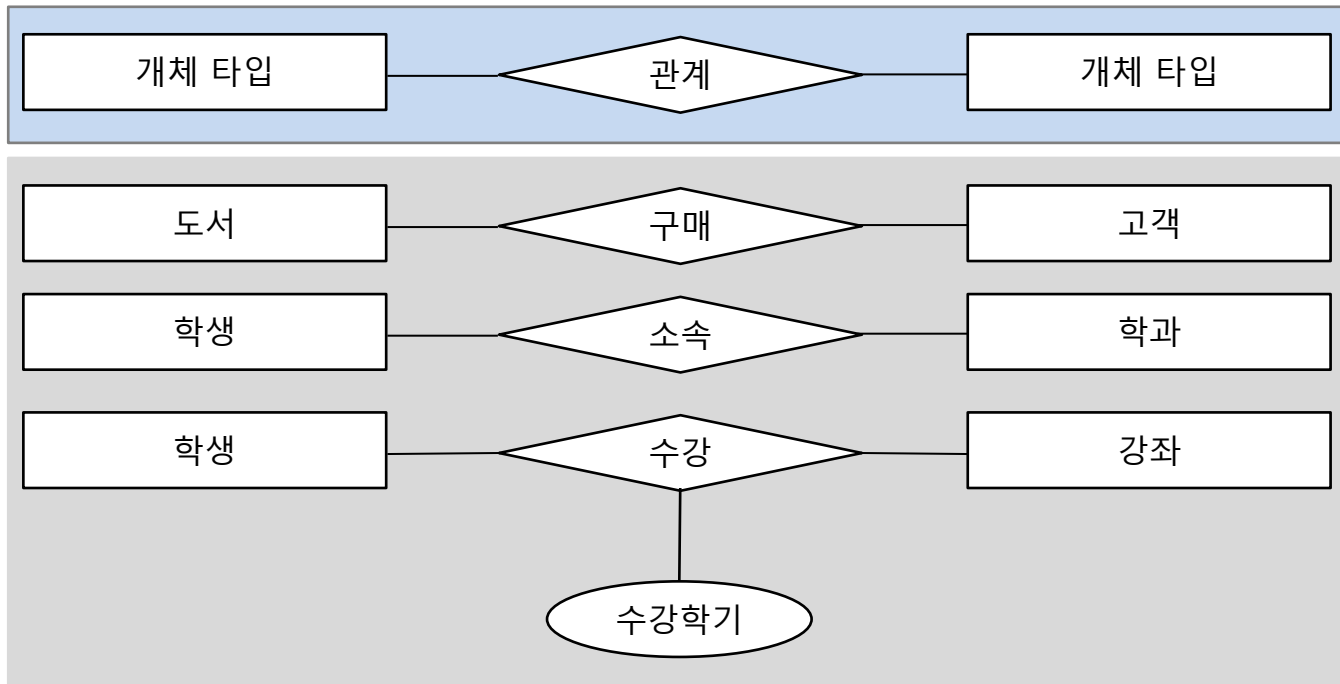



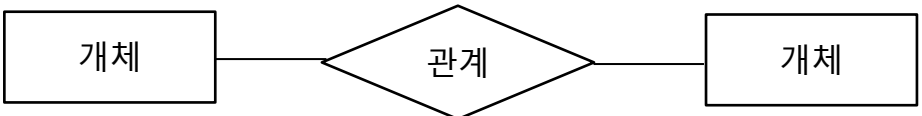
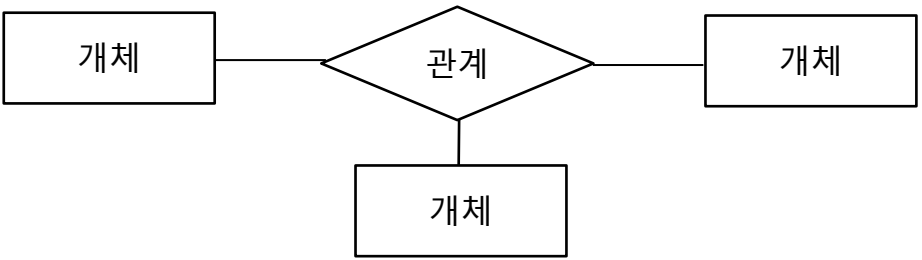
그림 6-16 관계의 예

## 2.3.2 관계 타입의 유형

### ■ 차수에 따른 유형

관계 집합에 참가하는 개체 타입의 수를 관계 타입의 차수(degree)라고 함.

표 6-5 차수에 따른 관계 타입의 유형

기호	의미	설명
	1진 관계	한 개의 개체가 자기 자신과 관계를 맺음
	2진 관계	두 개의 개체가 관계를 맺음
	3진 관계	세 개의 개체가 관계를 맺음

## 2.3.2 관계 타입의 유형

- 관계 대응수(cardinality) : 두 개체 타입의 관계에 실제로 참여하는 개별 개체 수

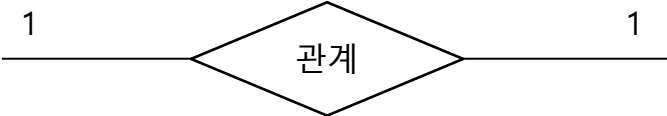
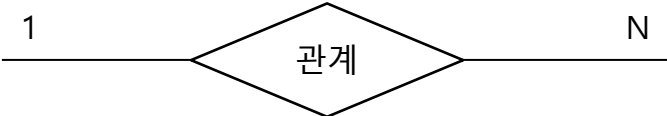
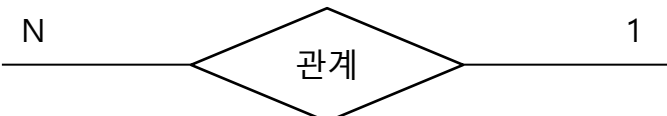
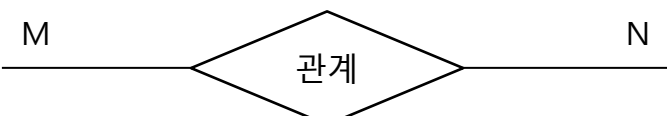
기호	의미	설명
	일대일 관계	하나의 개체가 하나의 개체에 대응
	일대다 관계	하나의 개체가 여러 개체에 대응
	다대일 관계	여러 개체가 하나의 개체에 대응
	다대다 관계	여러 개체가 여러 개체에 대응

표 6-6 관계 대응수에 따른 관계 타입의 유형

## 2.3.2 관계 타입의 유형

### ① 일대일(1:1)관계

좌측 개체 타입에 포함된 개체가 우측 개체 타입에 포함된 개체와 일대일로 대응하는 관계

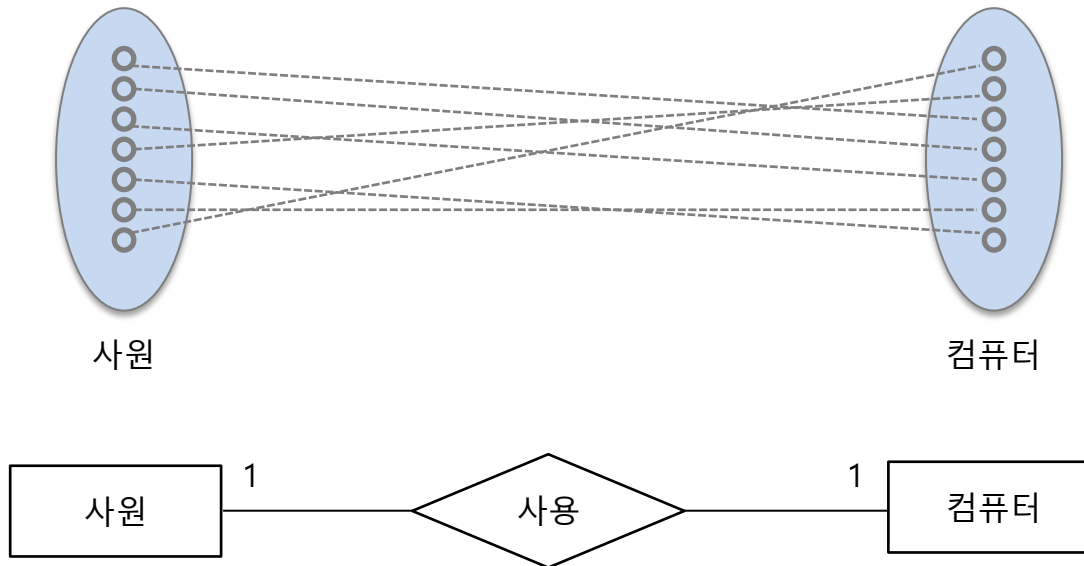


그림 6-20 일대일 관계의 예



## 2.3.2 관계 타입의 유형

### ② 일대다(1:N), 다대일(N:1) 관계

실제 일상생활에서 가장 많이 볼 수 있는 관계로, 한쪽 개체 타입의 개체 하나가 다른 쪽 개체 타입의 여러 개체와 관계를 맺음.

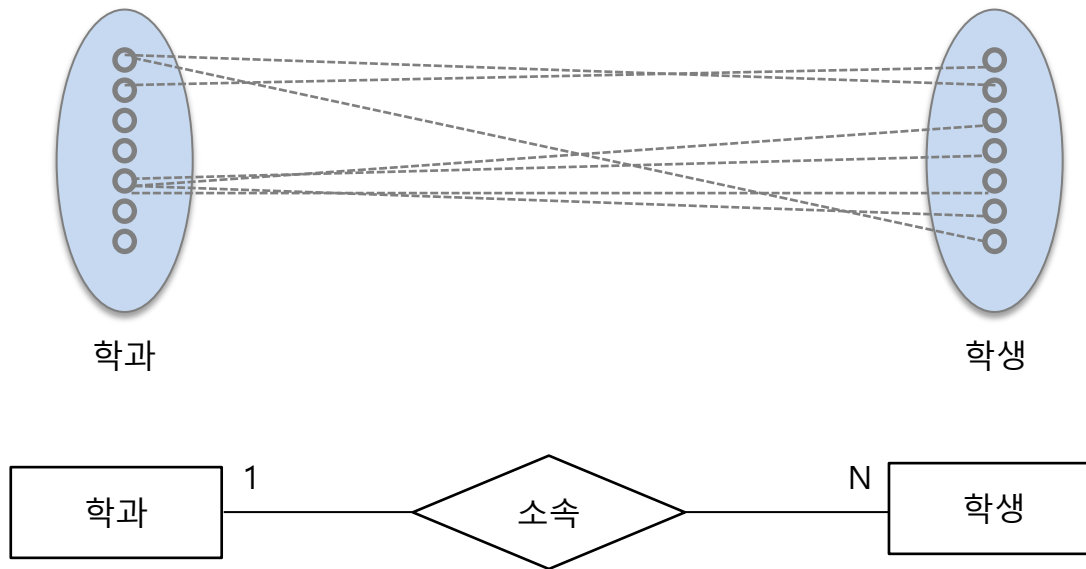


그림 6-21 일대다(1:N), 다대일(N:1) 관계의 예

## 2.3.2 관계 타입의 유형

### ③ 다대다(N:M) 관계

각 개체 타입의 개체들이 서로 임의의 개수의 개체들과 서로 복합적인 관계를 맺고 있는 관계를 말함.

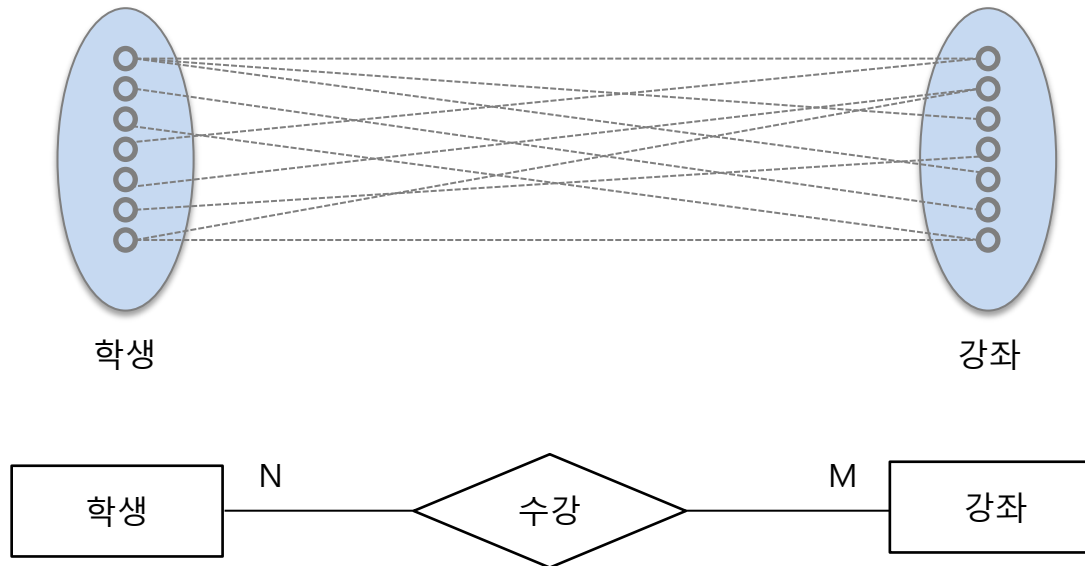


그림 6-22 다대다(N:M) 관계의 예

## 2.3.2 관계 타입의 유형

### ■ 관계 대응수의 최소값과 최대값

- 관계 대응수 1:1, 1:N, M:N에서 1, N, M은 각 개체가 관계에 참여하는 최대값을 의미함.
- 관계에 참여하는 개체의 최소값을 표시하지 않는다는 단점을 보완하기 위해 다이어그램에서는 대응수 외에 최소값과 최대값을 관계실선 위에 (최소값, 최대값)으로 표기함.

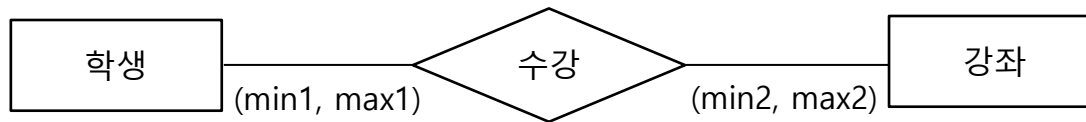


그림 6-23 관계 대응수의 최소값과 최대값의 표기

표 6-6 관계 대응수에 따른 관계 타입의 유형

관계	(min1,max1)	(min2,max2)
1:1	(0, 1)	(0, 1)
1:N	(0, *)	(0, 1)
M:N	(0, *)	(0, *)

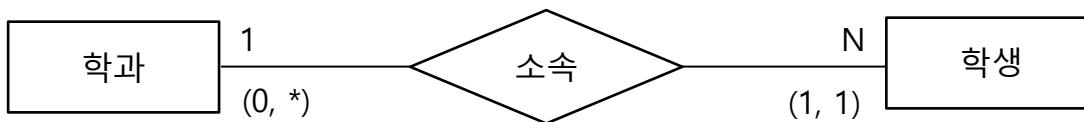
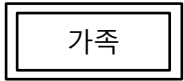
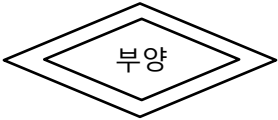
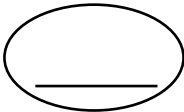
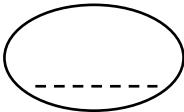


그림 6-24 (최소값, 최대값) 표기의 예

## 2.4 약한 개체 타입과 식별자

- 약한 개체(weak entity) 타입 : 상위 개체 타입이 결정되지 않으면 개별 개체를 식별할 수 없는 종속된 개체 타입
- 약한 개체 타입은 독립적인 키로는 존재할 수 없지만 상위 개체 타입의 키와 결합하여 약한 개체 타입의 개별 개체를 고유하게 식별하는 속성을 식별자(discriminator) 혹은 부분키(partial key)라고 함.

표 6-10 식별자와 약한 개체 타입

기호	의미	설명
	약한 개체 타입	<ul style="list-style-type: none"><li>• 강한 개체 타입이 있어야 존재할 수 있음</li><li>• 이중 직사각형으로 표현</li></ul>
	식별 관계 타입	<ul style="list-style-type: none"><li>• 강한 개체 타입과 약한 개체 타입의 관계를 나타냄</li><li>• 강한 개체 타입의 기본키를 상속받아 사용함</li><li>• 이중 마름모꼴로 표현</li></ul>
	키	<ul style="list-style-type: none"><li>• 강한 개체 타입의 키 속성</li></ul>
	식별자	<ul style="list-style-type: none"><li>• 약한 개체 타입에서 개별 개체를 구분하는 속성</li><li>• 키라고 하지 않고 식별자라고 부름</li></ul>

## 2.4 약한 개체 타입과 식별자

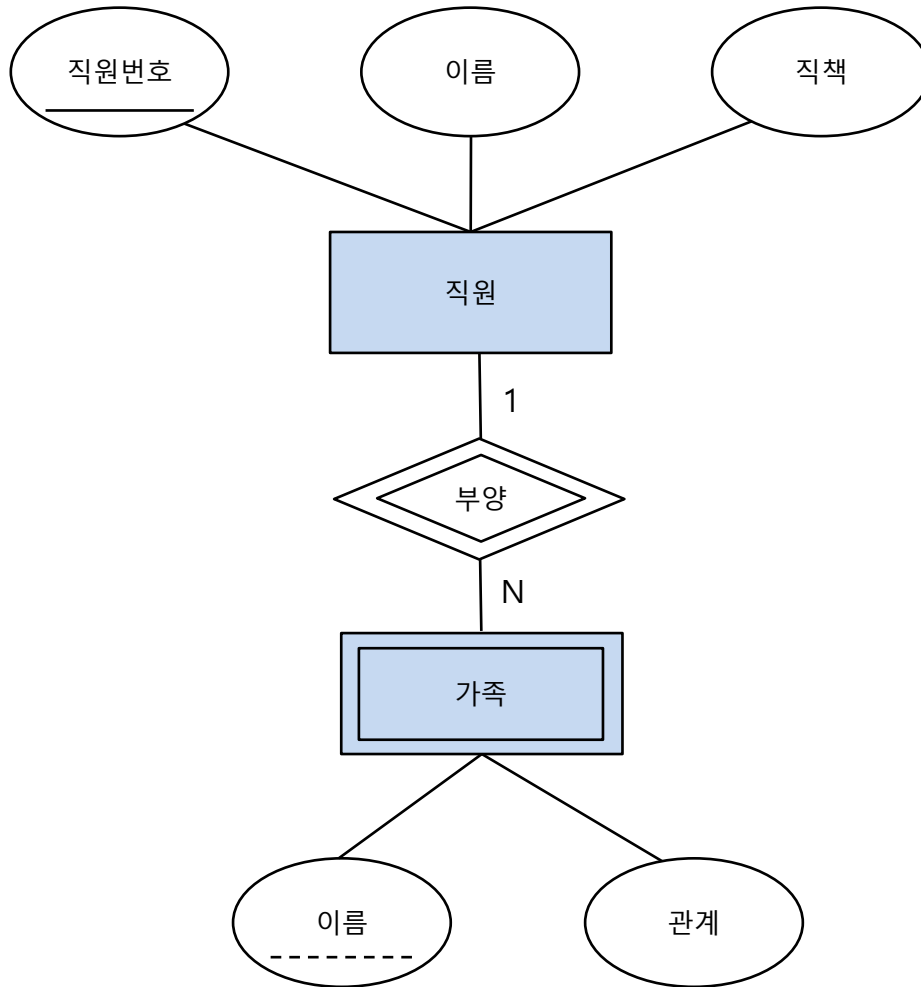


그림 6-29 약한 개체 타입과 식별자의 예

## 2.5 IE 표기법(Information Engineering 표기법)

- ER 다이어그램을 더 축약하여 쉽게 표현하면 eXERD 등 소프트웨어에서 사용함.
- IE 표기법에서 개체 타입과 속성은 직사각형으로 표현함.

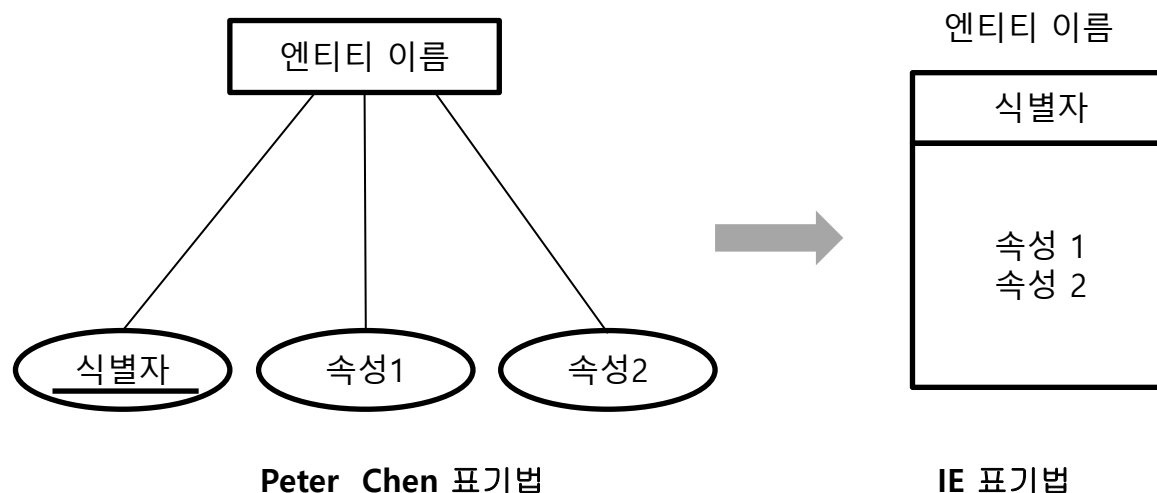


그림 6-31 Peter Chen 표기법과 IE 표기법

## 2.5 IE 표기법

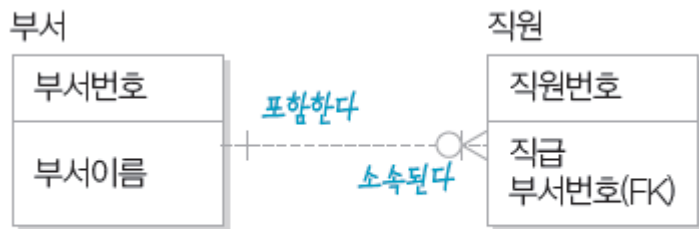
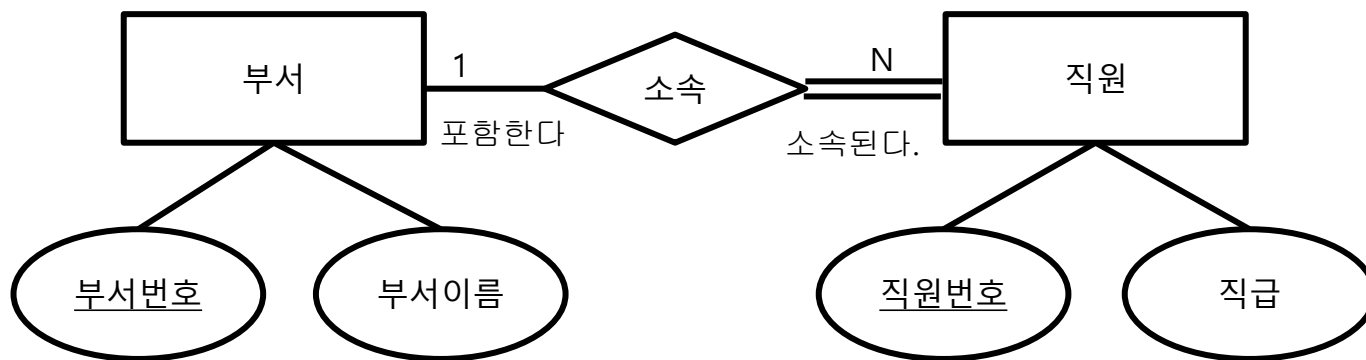
### ■ IE 표기법에서 관계는 실선 혹은 점선으로 표기함

그림 6-11 IE 표기법 - 관계와 관계 대응수

기호	의미
-----	<ul style="list-style-type: none"><li>• 비식별자 관계(non-identifying relationship): 강한 개체 타입</li><li>• 부모 개체의 키가 일반 속성으로 포함되는 관계</li></ul>
————	<ul style="list-style-type: none"><li>• 식별자 관계(identifying relationship): 약한 개체 타입</li><li>• 부모 개체의 키가 주식별자로 포함되는 관계</li></ul>
————<	<ul style="list-style-type: none"><li>• 일대다(1:N)의 관계: N 쪽에 새발을 표시</li></ul>
————○	<ul style="list-style-type: none"><li>• 0(선택 참여), 최소 참여가 0일 경우</li></ul>
————+	<ul style="list-style-type: none"><li>• 1(필수 참여), 최소 참여가 1일 경우</li></ul>

## 2.5 IE 표기법

- IE 표기법에서 관계(강한관계, 비식별자 관계)는 점선으로 표기함



(b) IE 표기법으로 작성한 직원-부서 관계

그림 6-31 IE 표기법의 예(비식별자 관계)

- IE 표기법에서 관계(약한관계, 식별자 관계)는 실선으로 표기함



그림 6-32 IE 표기법의 예(식별자 관계)



## 03. ER 모델을 관계 데이터 모델로 사상

---

- 개체 타입의 사상
- 관계 타입의 사상
- 다중 값 속성의 사상

### 03. ER 모델을 관계 데이터 모델로 사상

- 완성된 ER 모델은 실제 데이터베이스로 구축하기 위해 논리적 모델링 단계를 거치는데, 이 단계에서 사상(mapping)이 이루어짐.

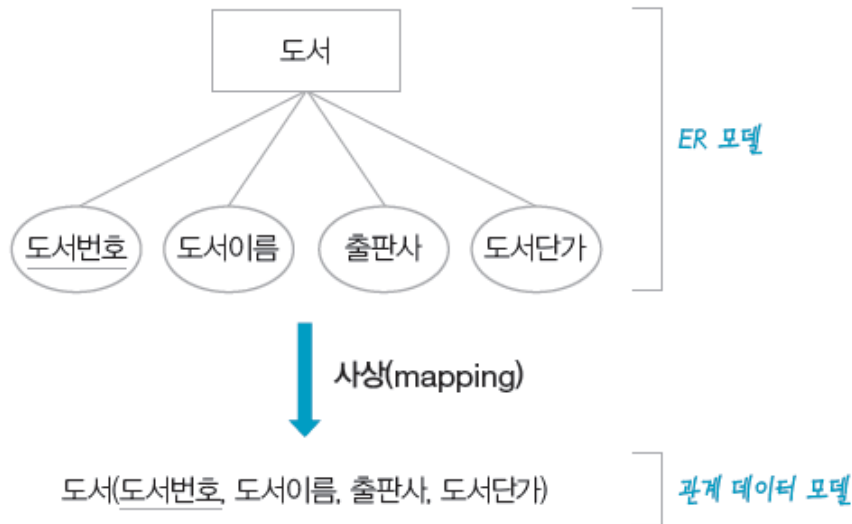


그림 6-33 ER 모델을 관계 데이터 모델로 사상

단계	사상할 대상	구분
1단계	개체 타입	강한 개체 타입
2단계		약한 개체 타입
3단계	관계 타입	이진 1:1 관계 타입
4단계		이진 1:N 관계 타입
5단계		이진 N:M 관계 타입
6단계		N진 관계 타입
7단계	속성	다중값 속성

표 6-12 ER 모델과 관계 데이터 모델의 사상 알고리즘

## 3.1 개체 타입의 사상

- [1단계] 강한(정규) 개체 타입 정규 개체 타입 E의 경우 대응하는 릴레이션 R을 생성함.
- [2단계] 약한 개체 타입 : 약한 개체 타입에서 생성된 릴레이션은 자신의 키와 함께 강한 개체 타입의 키를 외래키로 사상하여 자신의 기본키를 구성함.

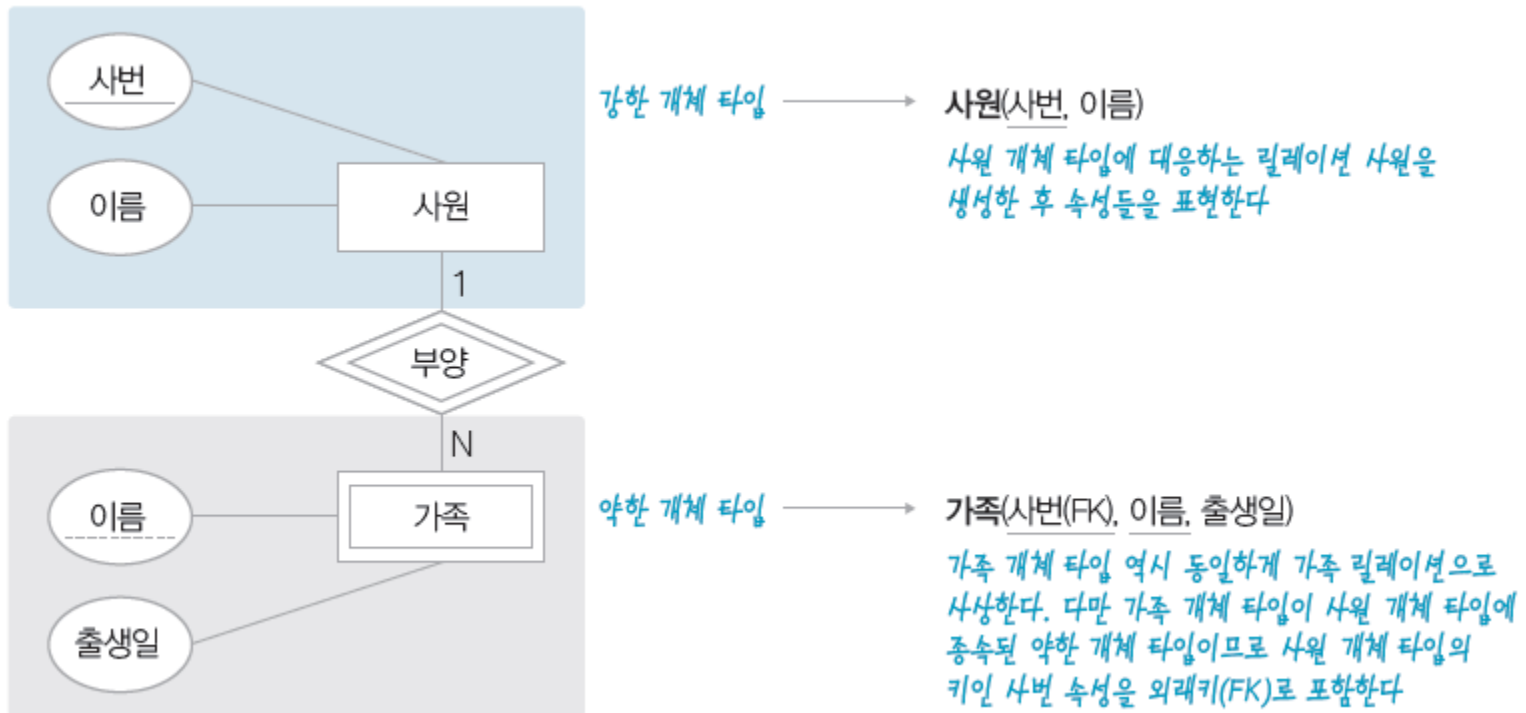


그림 6-34 개체 타입의 사상

## 3.2 관계 타입의 사상

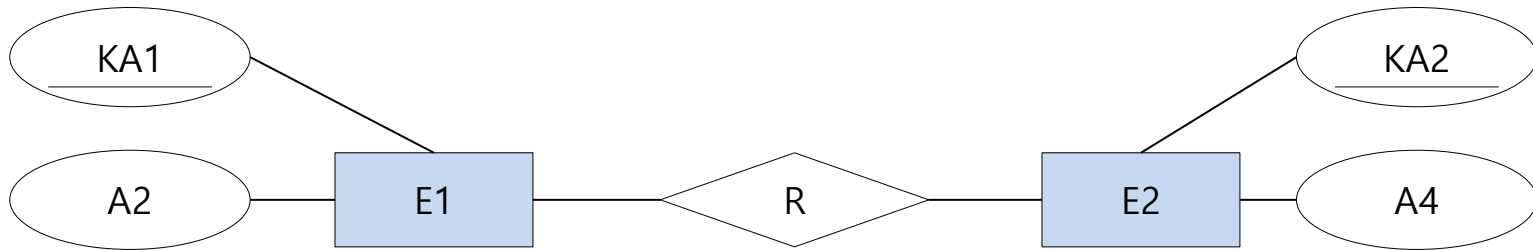


그림 6-34 이진 관계 타입

**[방법1]** 오른쪽 개체 타입 E2를 기준으로 관계 R을 표현한다.

E1(KA1, A2)

E2(KA2, A4, KA1)

**[방법2]** 왼쪽 개체 타입 E1을 기준으로 관계 R을 표현한다.

E1(KA1, A2, KA2)

E2(KA2, A4)

**[방법3]** 단일 릴레이션 ER로 모두 통합하여 관계 R을 표현한다.

ER(KA1, A2, KA2, A4)

**[방법4]** 개체 타입 E1, E2와 관계 타입 R을 모두 독립된 릴레이션으로 표현한다.

E1(KA1, A2)

R(KA1, KA2)

E2(KA2, A4)

## 3.2 관계 타입의 사상

### ■ [3단계] 이진 1:1 관계 타입

이진 1:1 관계 타입의 경우 [방법1]~[방법4]까지 모든 유형으로 사상 가능. 개체가 가진 정보 유형에 따라 판단.

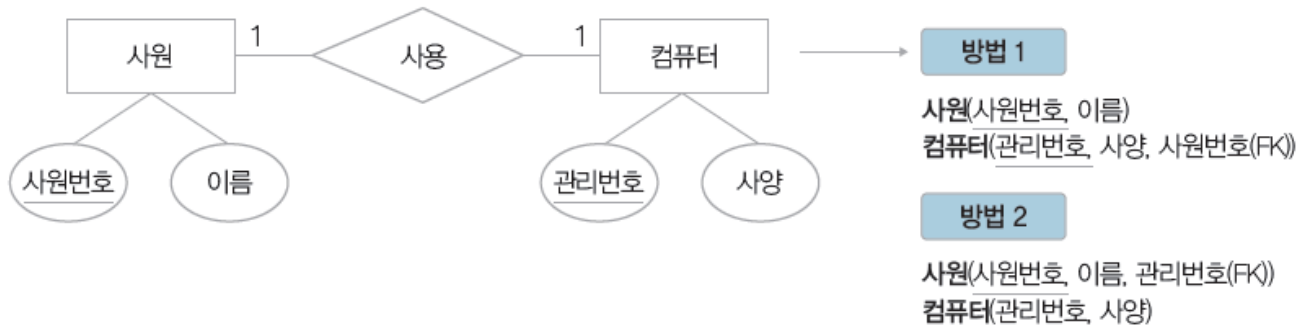


그림 6-36 이진 1:1 관계 타입의 사상

### ■ [4단계] 이진 1:N 관계 타입

이진 1:N 관계 타입의 경우 N의 위치에 따라 [방법1] 또는 [방법2]의 유형으로 사상됨.

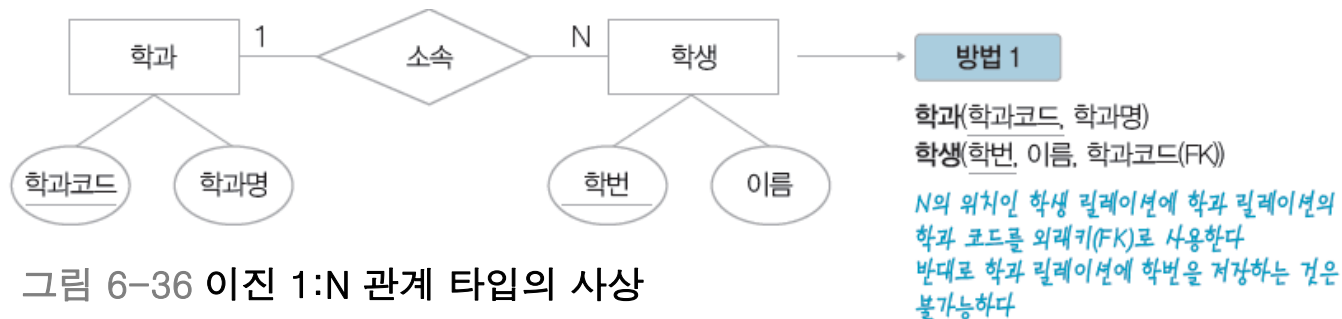


그림 6-36 이진 1:N 관계 타입의 사상

## 3.2 관계 타입의 사상

### ■ [5단계] 이진 M:N 관계 타입

이진 M:N 관계 타입은 [방법4]의 유형으로 사상됨.

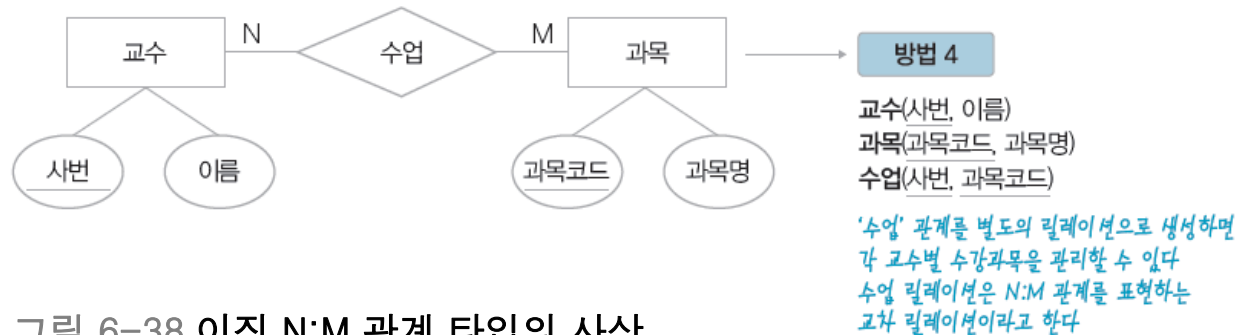


그림 6-38 이진 N:M 관계 타입의 사상

### ■ [6단계] N진 관계 타입

ER 모델의 차수가 3 이상인 다진 관계 타입의 경우 [방법4]의 유형으로 사상된다.

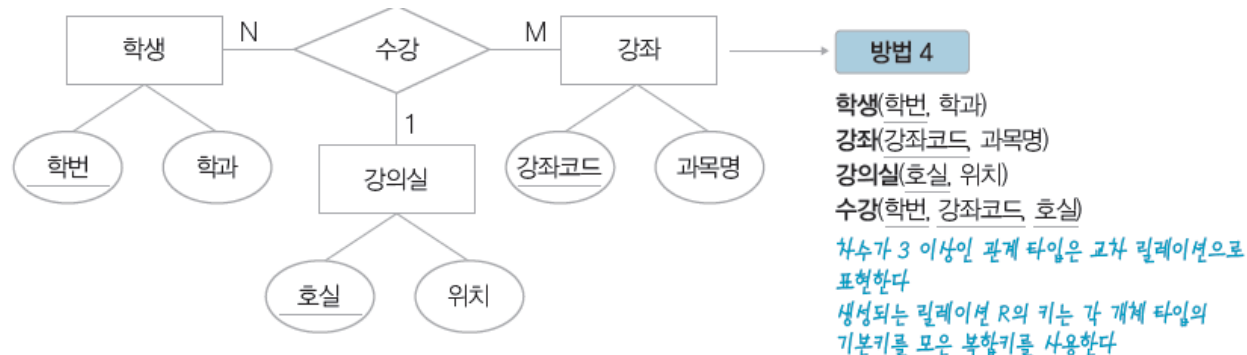


그림 6-36 이진 N진 관계 타입의 사상

### 3.3 다중 값 속성의 사상



그림 6-35 다중 값 속성의 개수에 따른 사상 방법

#### ■ [7단계]

속성의 개수를 알 수 없는 경우 [방법1]을, 속성의 개수가 제한적으로 정해지는 경우 [방법2]를 사용함.

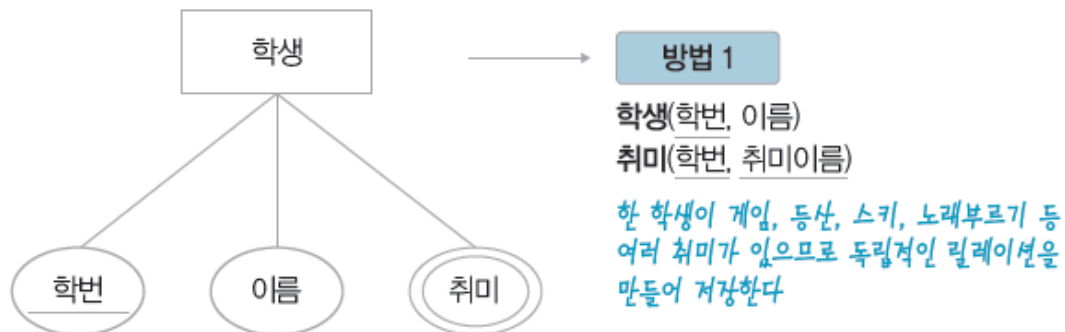


그림 6-41 다중 값 속성의 사상

## 04. eXERD 실습

---

- eXERD 기본 화면 및 툴 둘러보기

<http://ko.exerd.com/#download-section>

- eXERD 실습을 위한 기본 환경 설정하기
- 마당서점 설계 실습
- DBMS에 접속하여 테이블 생성하기



## 04. eXERD 실습

- eXERD : 데이터 모델링을 하기 위한 프로그램. IE 표기법을 지원함.

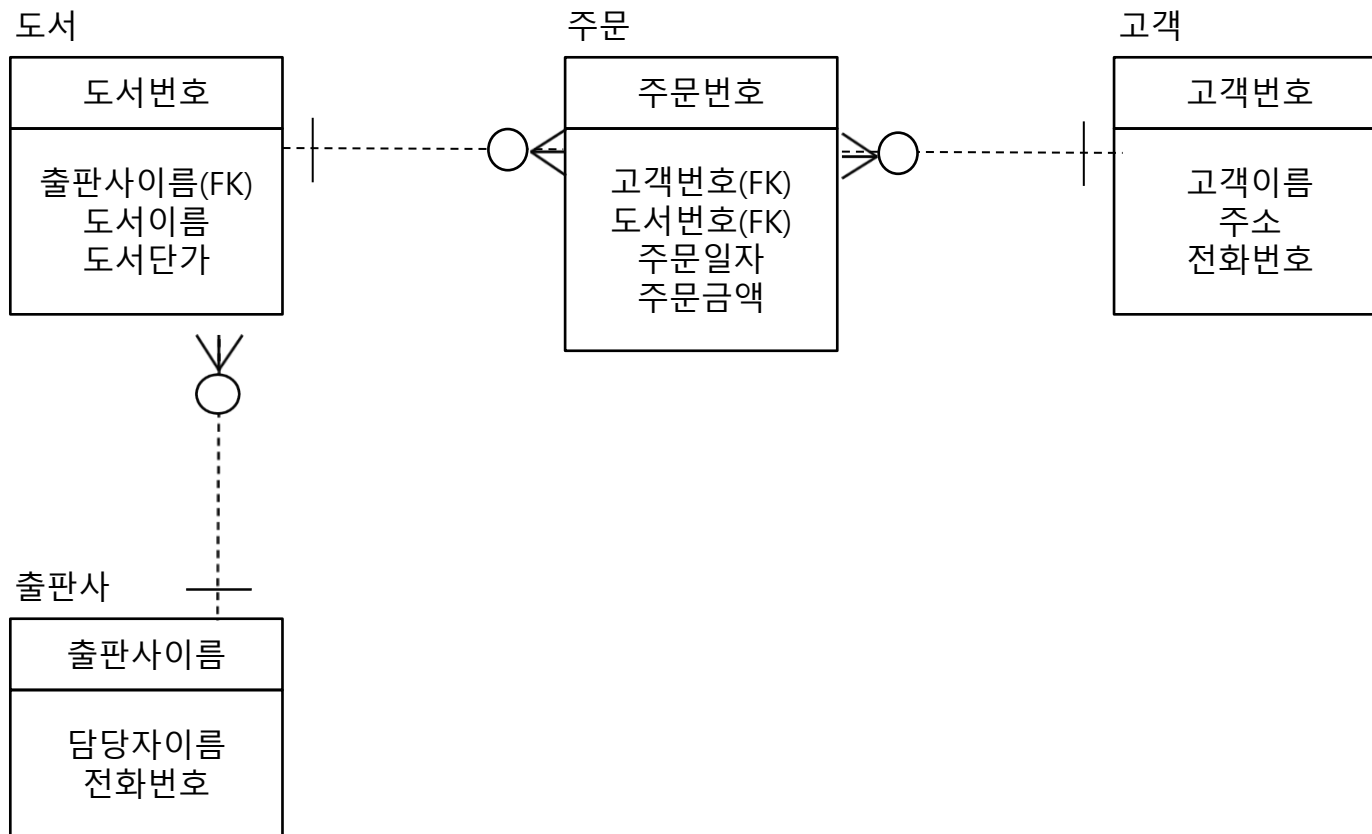


그림 6-36 마당서점의 ER 다이어그램

# 4.1 eXERD 기본 화면 및 툴 둘러보기

The screenshot displays the eXERD software interface with the following components:

- Top Menu Bar:** 파일(F), 편집(E), 탐색(N), 검색(A), 프로젝트(P), 실행, eXERD, 창(W), 도움말(H).
- Left Panel:**
  - 네비게이터:** Thumbnail view of the diagram.
  - Palette:** default, 선택 툴, 패닝, 생성 도구, 테이블, 비식별 관계, 식별 관계, 노트.
  - 모델:** 새 데이터베이스: Oracle 9i/10g/11g, 수강신청.
  - 히스토리:** study.exerd, 개정 시간: 12. 4. 24. 오후 2:25, 12. 4. 23. 오후 5:19, 12. 4. 23. 오후 5:19.
- Central Canvas:** Diagram showing tables: LECTURE\_ROOM (강의실), LECTURE\_ROOM\_ASSIGN (강의실배정), and SUBJECT (교과목). Relationships are indicated by lines. A large play button is overlaid on the diagram.
- Right Panel:**
  - study.exerd 현재(와) 로컬 개정 비교:** Table with columns 위치 and 상세 내용.

위치	상세 내용
종료시간	컬럼이 제거됨
컬럼 세부속성	컬럼 세부속성이 제거됨
새 컬럼	컬럼이 추가됨
  - 로컬: study.exerd / 로컬 히스토리: stud...3. 오후 5:19:07:** Tree view showing the database structure: 다이어그램, 문서 설정, 데이터베이스, 데이터 사전, 수강신청, 교시, 컬럼, 교시코드, 시작 시간, 종료 시간, 컬럼 그룹, 테이블 세부속, 강의실, 강의실배정.
  - 검토:** List of messages:

설명	타입	우
6개의 개체 이름이 지정되지 않았습니다.	이름 지정되지 않음	
등록일시의 물리명은 RGSTN_DTO이어야 합니다.	용어 정책 위반	
새 테이블(TABLE) 테이블의 물리 이름에 예약어 TABLE이 사용되었	이름에 예약어 사용	
이미지 도메인의 데이터 타입이 올바르지 않습니다.	올바르지 않은 데이터 타입	
LECTURE_ROOM_TO_LECTURE_ROOM_ASSIGN_FK은 너무 긴 이름	너무 긴 물리명	

## 4.3 마당서점 설계 실습

### ■ 1. 마당서점의 논리적 모델링

#### ① 마당서점의 요구사항 분석 후 개체 만들기



그림 6-52 출판사 개체 생성

#### ② 개체 간 관계 표현하기

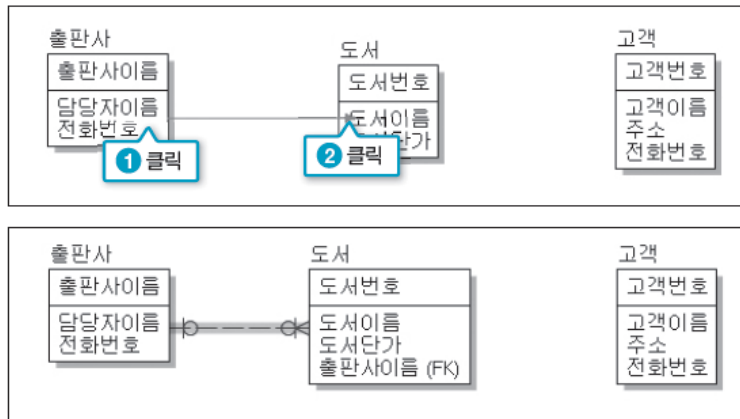


그림 6-53  
출판사, 도서 개체의 관계 설정(1:N 비식별)

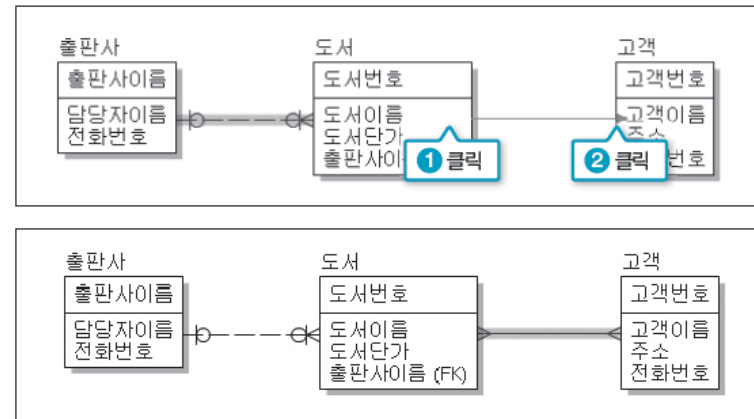


그림 6-54  
도서, 고객 개체의 관계 설정(N:M 관계)

## 4.3.2 도메인 정의하기

- 도메인이란 속성이 가질 수 있는 값을 정의하는 것. ER 다이어그램이 완성 후 도메인을 정의함.

구분(기본도메인)	도메인	데이터 타입
숫자(Number)	번호	NUMBER
	금액	NUMBER
	단가	NUMBER
문자(String)	이름	VARCHAR(40)
	주소	VARCHAR(40)
	전화번호	VARCHAR(30)
	일자	DATE
날짜 · 시간(Datetime)	일자	DATE

표 6-17 마당서점의 도메인별 데이터 타입 정의표

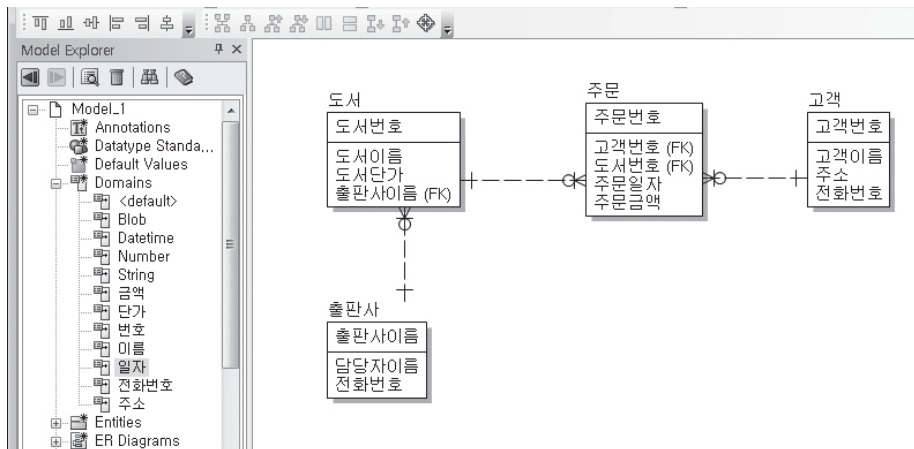


그림 6-60 도메인 정의표에 따라 생성한 도메인

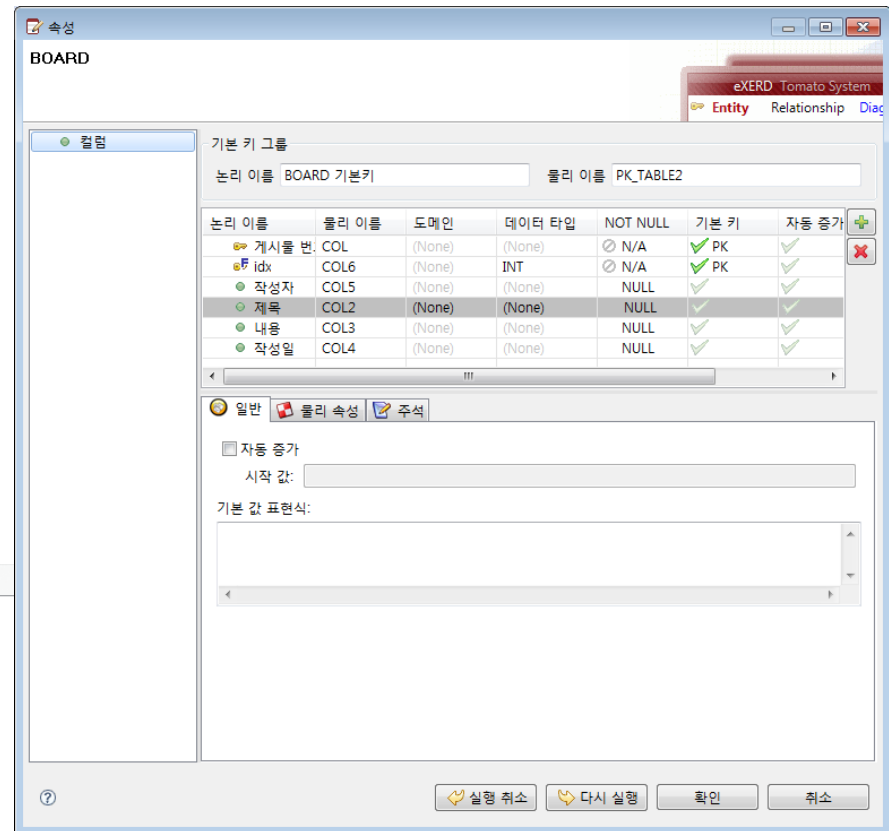


그림 6-61 도메인별 데이터 타입 설정

## 4.3.3 마당서점의 물리적 모델링

### ① 물리적 모델링

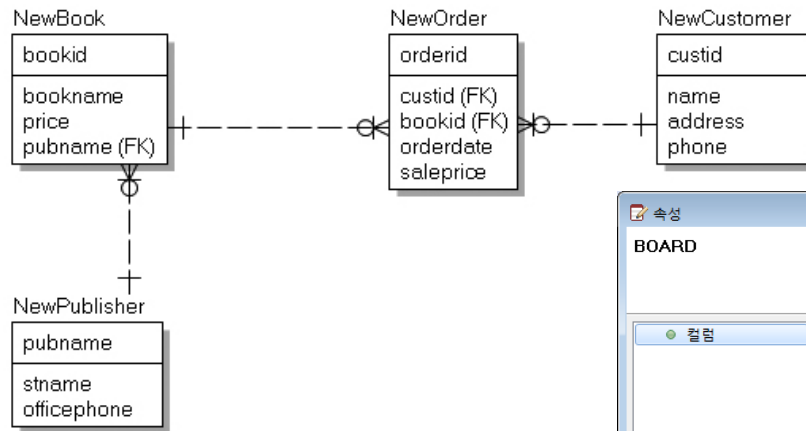


그림 6-64 마당서점의 테이블

### ② 컬럼의 속성 확인하기

The screenshot shows the eXRD Tomato System interface for the BOARD table. The '속성' (Properties) window is open, displaying the '컬럼' (Columns) tab. The table name is 'BOARD' and the physical name is 'PK\_TABLE2'. The columns are listed in a table with the following details:

논리 이름	물리 이름	도메인	데이터 타입	NOT NULL	기본 키	자동 증가
게시물 번	COL	(None)	(None)	N/A	PK	사용
idx	COL6	(None)	INT	N/A	PK	사용
작성자	COL5	(None)	(None)	NULL		
제목	COL2	(None)	(None)	NULL		
내용	COL3	(None)	(None)	NULL		
작성일	COL4	(None)	(None)	NULL		

Below the table, there are tabs for '일반' (General), '물리 속성' (Physical Properties), and '주석' (Comments). The '일반' tab is selected, showing options for '자동 증가' (Auto Increment) and '기본 값 표현식' (Default Value Expression).

그림 6-65 컬럼 속성 확인