

데이터베이스 모델링

문혜영

학습 목차

- 01 데이터 모델링
- 02 개체-관계 모델
- 03 논리적 데이터 모델

- 데이터 모델링(data modeling)

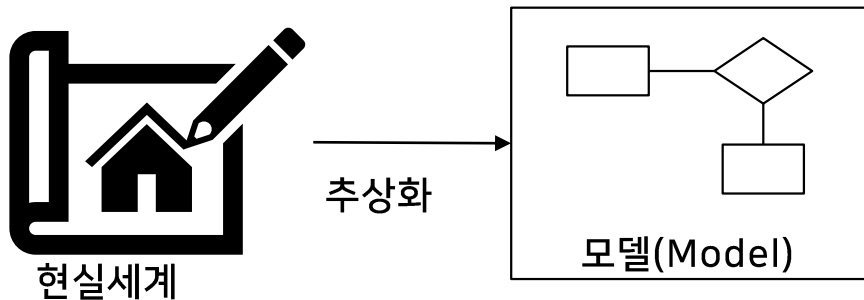
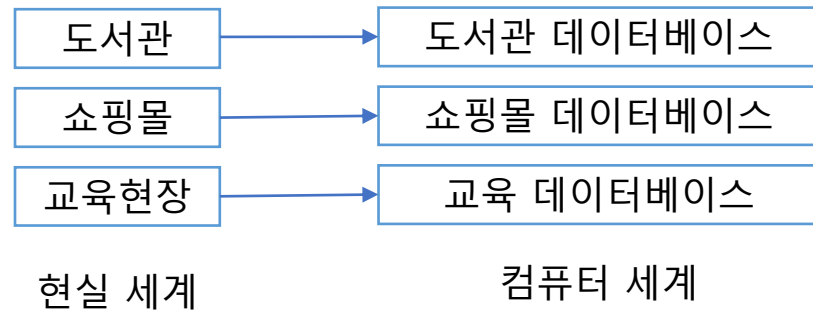
- 현실세계의 복잡한 정보와 관계를 컴퓨터 세계에서 처리할 수 있도록 변환하는 과정

- 추상화 (↔ 구체화)

- 도서관 시스템 : 도서, 회원, 대출, 예약

- 온라인 쇼핑몰 : 상품, 고객, 주문, 리뷰, 재고

- 교육 시스템 : 학생, 강사, 과목, 성적, 시간표



```
click,close-full-overlay",...
review"),render:function()...
ter,navigate(c.router,bu...
$.el.addClass("iframe-...
removeClass("iframe-...
```

- 모델링의 세가지 관점
 - 데이터 관점
 - What, 무엇이 필요한지
 - 데이터와 데이터 간 관계, 업무와 데이터 간 관계를 모델링
 - 데이터에 접근하는 방법(How), 사람(Who)과는 무관
 - 프로세스 관점(How)
 - 업무가 실제로 하고 있는 일 또는 해야할 일을 모델링
 - 데이터와 프로세스의 상관 관점(Interaction)
 - 업무 처리 방법에 따라 데이터가 받는 영향을 모델링



- 데이터 모델링의 3단계

- 1. 개념적 모델링(Conceptual Data Modeling):

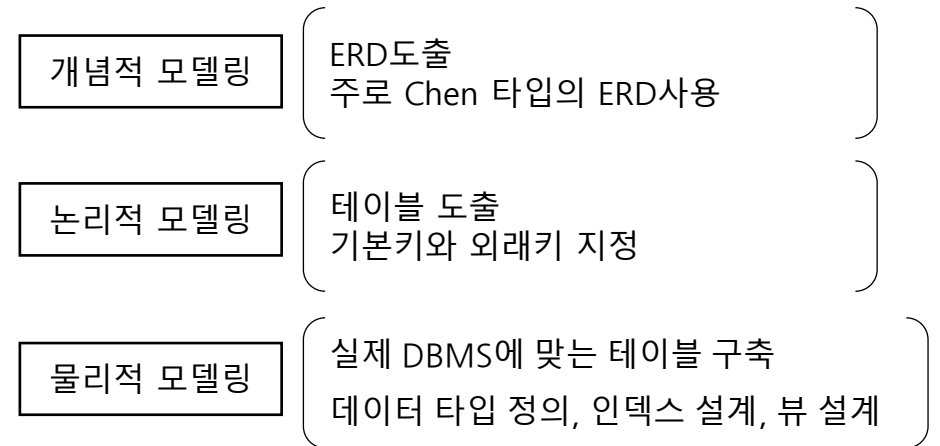
- 의사소통을 돕고, 전체적인 데이터 흐름과 구조에 대한 이해를 높이는 목적
 - 예) 개체-관계 모델

- 2. 논리적 모델링(Logical Data Modeling):

- 개념적 모델을 좀 더 구체화하여, 데이터의 논리적 구조와 관계를 정의
 - 데이터의 타입, 기본 키, 외래 키, 인덱스 등을 명세
 - 예) 관계 데이터 모델

- 3. 물리적 모델링(Physical Data Modeling):

- 논리적 모델을 실제 데이터베이스 구현에 맞게 최적화
 - 데이터의 저장, 액세스 방식에 대한 세부 사항을 정리





- 예시 : 온라인 쇼핑몰

- 1) 개념적 모델링 (Conceptual Modeling)

- 온라인 쇼핑몰의 주요 엔터티(Entity)와 그들 간의 관계(Relationship)를 정의
 - 주요 엔터티: 고객, 상품, 주문, 리뷰 등
 - 주요 관계
 - 고객은 주문을 할 수 있다.
 - 주문은 여러 상품을 포함할 수 있다.
 - 상품은 여러 리뷰를 가질 수 있다.
 - 리뷰는 한 명의 고객에 의해 작성될 수 있다.
 - 다양한 ERD(Entity-Relationship Diagram) 표기법으로 표현
 - Peter Chen 표기법, IE 표기법, Barker 표기법, ...



- 예시 : 온라인 쇼핑몰

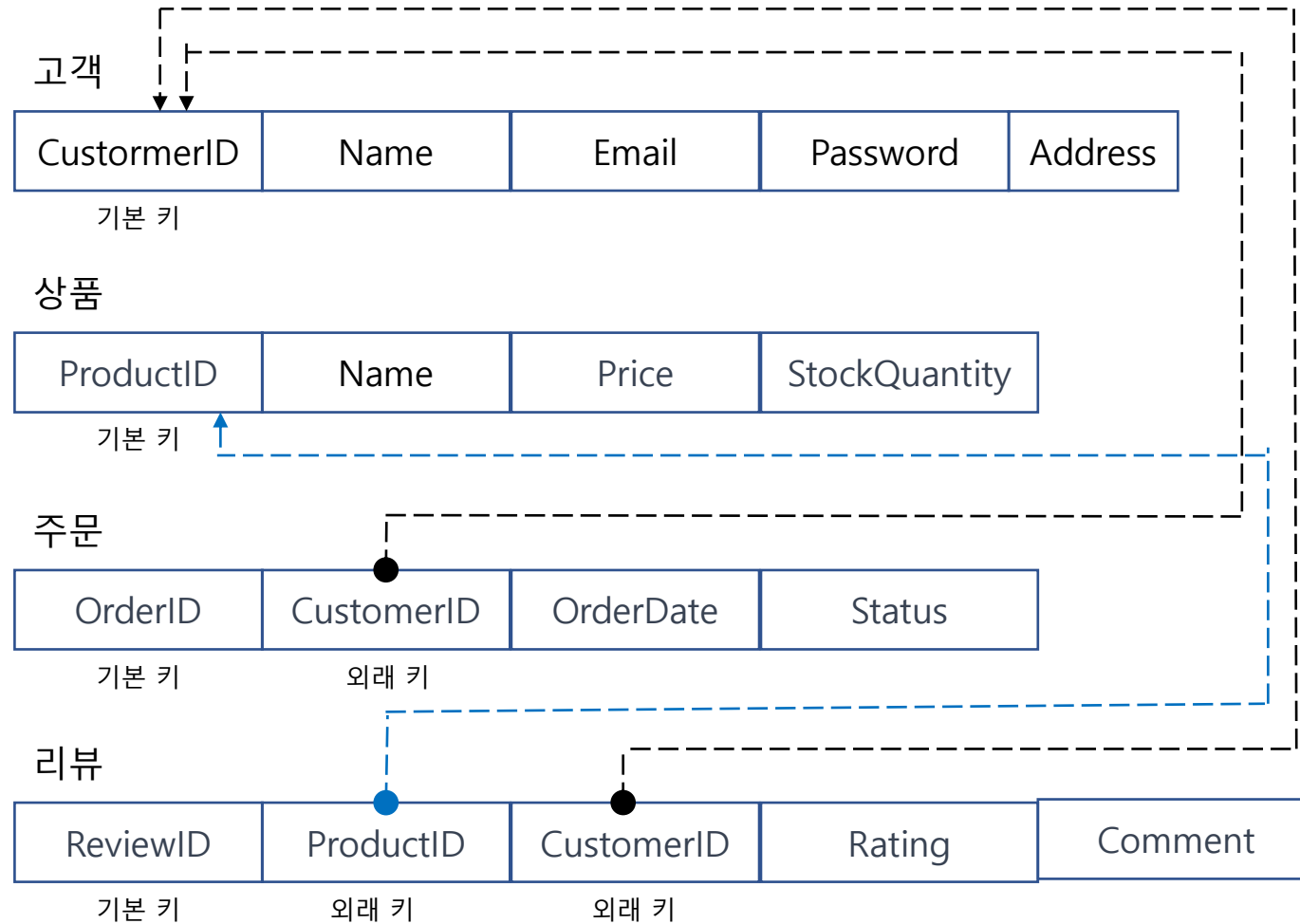
- 2) 논리적 모델링 (Logical Modeling)

논리적 모델링에서는 개념적 모델을 좀 더 구체화

- 고객 엔터티 : CustomerID (기본 키), Name, Email, Password, Address
 - 상품 엔터티 : ProductID (기본 키), Name, Price, StockQuantity
 - 주문 엔터티 : OrderID (기본 키), CustomerID (외래 키), OrderDate, Status
 - 리뷰 엔터티 : ReviewID (기본 키), ProductID (외래 키), CustomerID (외래 키), Rating, Comment
 - 관계와 제약 조건:
 - 한 명의 고객은 여러 주문을 할 수 있음 (고객 1:N 주문)
 - 한 개의 주문은 여러 상품을 포함할 수 있음 (주문 1:N 상품)
 - 하나의 상품은 여러 리뷰를 가질 수 있음 (상품 1:N 리뷰)
 - 하나의 리뷰는 하나의 고객에 의해 작성됨 (고객 1:1 리뷰)

01 데이터 모델링

- 온라인 쇼핑몰
 - 2) 논리적 모델링




```
click,close-full-overlay" data-bbox="700 0 975 115">review"),render:function()var
ter,navigate(c.router,basename
$.addClass("iframe-ready")
removeClass("iframe-ready")
```

- 예시 : 동물 보호소

- 1) 개념적 모델링 (Conceptual Modeling)

- 동물 보호소의 주요 엔터티(Entity)와 그들 간의 관계(Relationship)
 - 주요 엔터티: 동물, 입양인, 치료 기록, 입양 기록 등
 - 주요 관계:
 - 동물은 하나의 보호자에 의해 관리될 수 있다.
 - 입양인은 여러 동물을 입양할 수 있다.
 - 동물은 치료 기록을 여러 개 가질 수 있다.
 - 입양 기록은 특정 동물과 입양인에 연결될 수 있다.



- 예시 : 동물보호소

- 2) 논리적 모델링 (Logical Modeling)

- 개념적 모델을 더 구체적으로 만들어 감

- 동물 엔터티 : AnimalID (기본 키), Name, Species, Age, HealthStatus
 - 입양인 엔터티 : AdopterID (기본 키), Name, Address, Phone
 - 치료기록 엔터티 : TreatmentID (기본 키), AnimalID (외래 키), TreatmentDate, Details
 - 입양기록 엔터티 : AdoptionID (기본 키), AnimalID (외래 키), AdopterID (외래 키), AdoptionDate

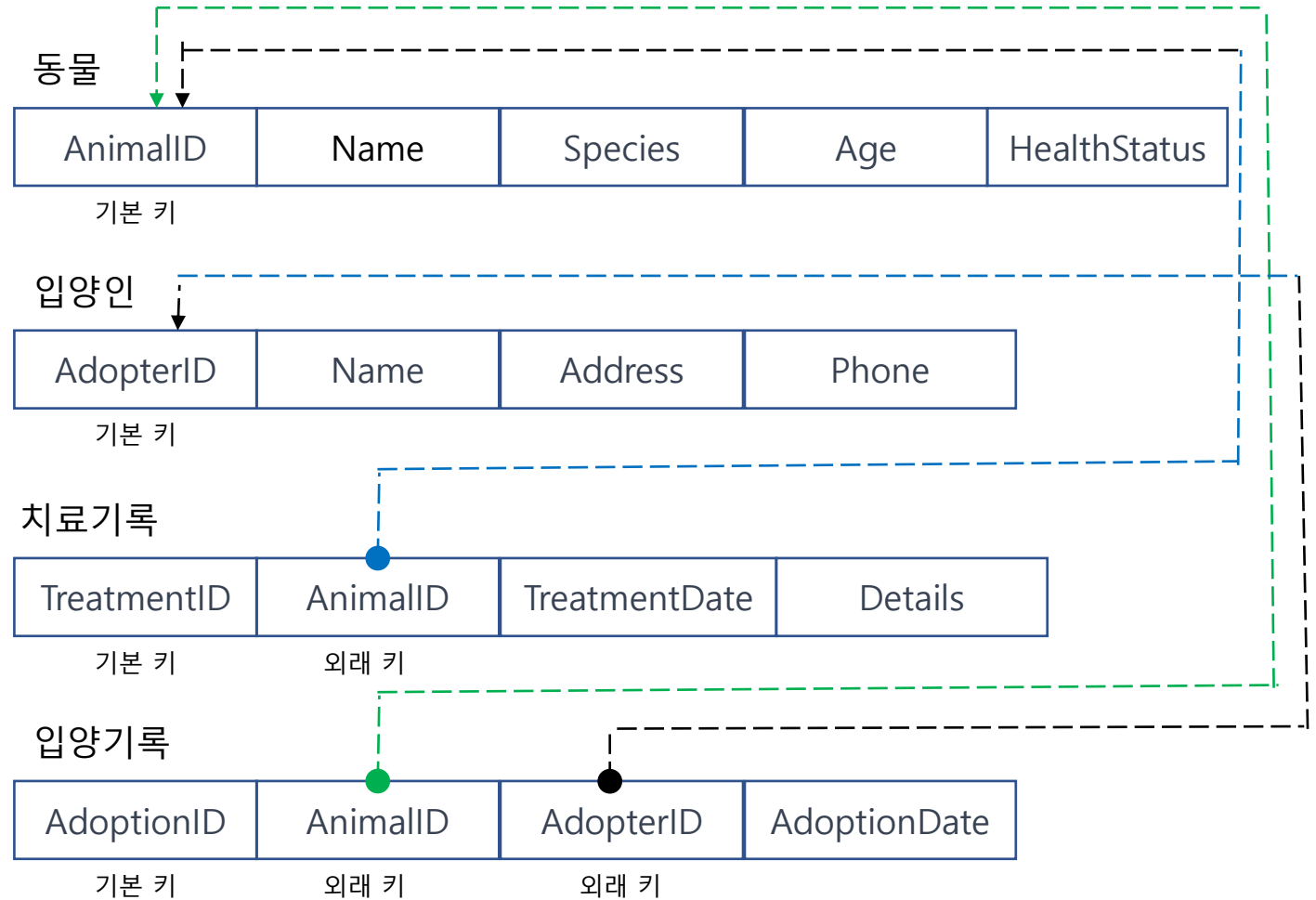
- 관계와 제약 조건:

- 한 명의 입양인은 여러 동물을 입양할 수 있다. (입양인 1:N 동물)
 - 한 마리의 동물은 여러 치료 기록을 가질 수 있다. (동물 1:N 치료 기록)
 - 입양 기록은 특정 동물과 입양인에 연결된다. (동물 1:1 입양 기록, 입양인 1:N 입양 기록)

01 데이터 모델링

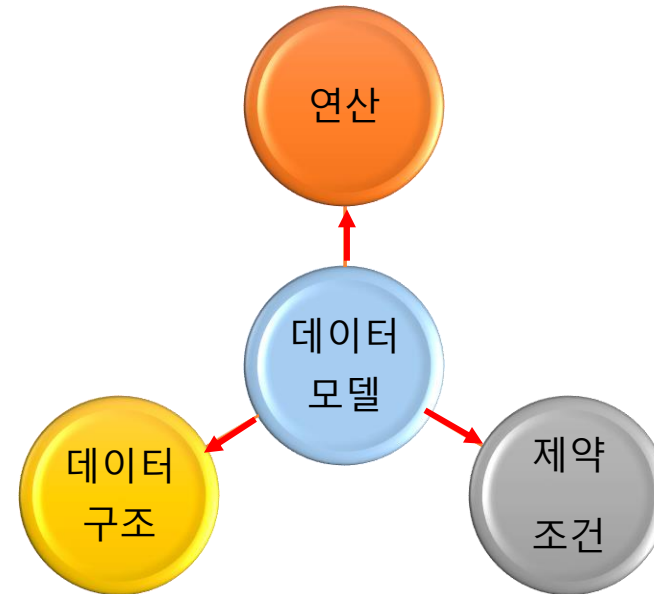
- 예시 : 동물보호소

- 2) 논리적 모델링 (Logical Modeling)



- 데이터 모델

- 데이터 모델링의 결과물을 표현하는 도구
- 데이터 모델에 표시할 요소
 - 데이터 구조(data structure)
 - 구성 요소
 - 정적 특징
 - 연산(operation)
 - 표현된 값들을 처리하는 작업
 - 동적 특징
 - 제약조건(constraint)
 - 데이터 무결성 유지 목적



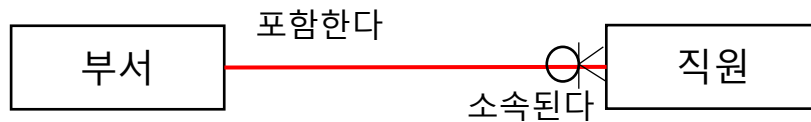
```
click.closeFullOverlay()
review"),render:function()
ter.navigate(c.router.basem
$.addClass("iframe-root")
removeClass("iframe-root")
```

- 데이터 모델 표시법

- Chen



- IE/Crow's Foot



- 기타

- IDEF1X (Integration Definition for Information Modeling)
 - UML (Unified Modeling Language)
 - :

```
click.close-full-overview()
review"),render:function() {
ter.navigate(c.router, {back:
$.addClass("iframe-resize-handle")
removeClass("iframe-resize-handle")
```

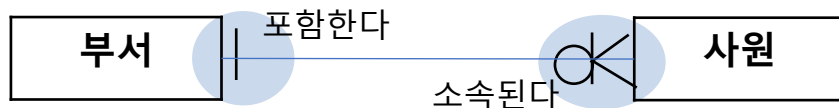
- 개체-관계 모델(ERM, E-R Model, Entity-Relationship model)
 - 피터 첸(Peter Chen)이 제안한 개념적 데이터 모델
 - 개체와 개체 간의 관계를 이용해 현실 세계를 개념적 구조로 표현
 - 핵심 요소 : 개체, 속성, 관계
 - Entity(개체): 데이터 모델에서 대상이나 개념
 - Attribute(속성): 개체의 특성이나 성질
 - Relationship(관계): 개체간의 연관성이나 상호작용
- 개체-관계 다이어그램(ERD, E-R diagram, Entity-Relationship diagram)
 - 개체-관계 모델을 시각적으로 표현-> 관계자들이 쉽게 이해할 수 있도록 함



```
click.closeFullOverlay()
review"),render:function() {
ter.navigate(c.router, {name:
$.addClass("iframe-ready");
removeClass("iframe-ready");
```

- ERD 작성 순서

- 엔티티를 그린 후 적절하게 배치
 - 가급적 선이 꼬이지 않게 배치
 - 왼쪽 → 오른쪽, 위 → 아래 순으로 읽어 나가기 편하도록 배치
- 엔티티간 관계 설정
 - 식별자 관계를 우선 설정함
 - 가급적 Cycle 관계도 발생하지 않아야 함
- 관계명 기술 (양 방향)
 - 현재형 사용, 지나치게 포괄적인 단어는 지양
 - 실제 프로젝트에서는 크게 고려하지 않음
- 관계차수와 선택성 표시



02 개체-관계 모델(개체)

- 엔터티(Entity)의 정의

- 변별할 수 있는 사물 - Peter Chen (1976)
- 데이터베이스 내에서 변별 가능한 객체 - C.J Date (1986)
- 정보를 저장할 수 있는 어떤 것 - James Martin (1989)
- 정보가 저장될 수 있는 사람, 장소, 물건, 사건 그리고 개념 등 - Thomas Bruce (1992)



의사



말



사건



과목

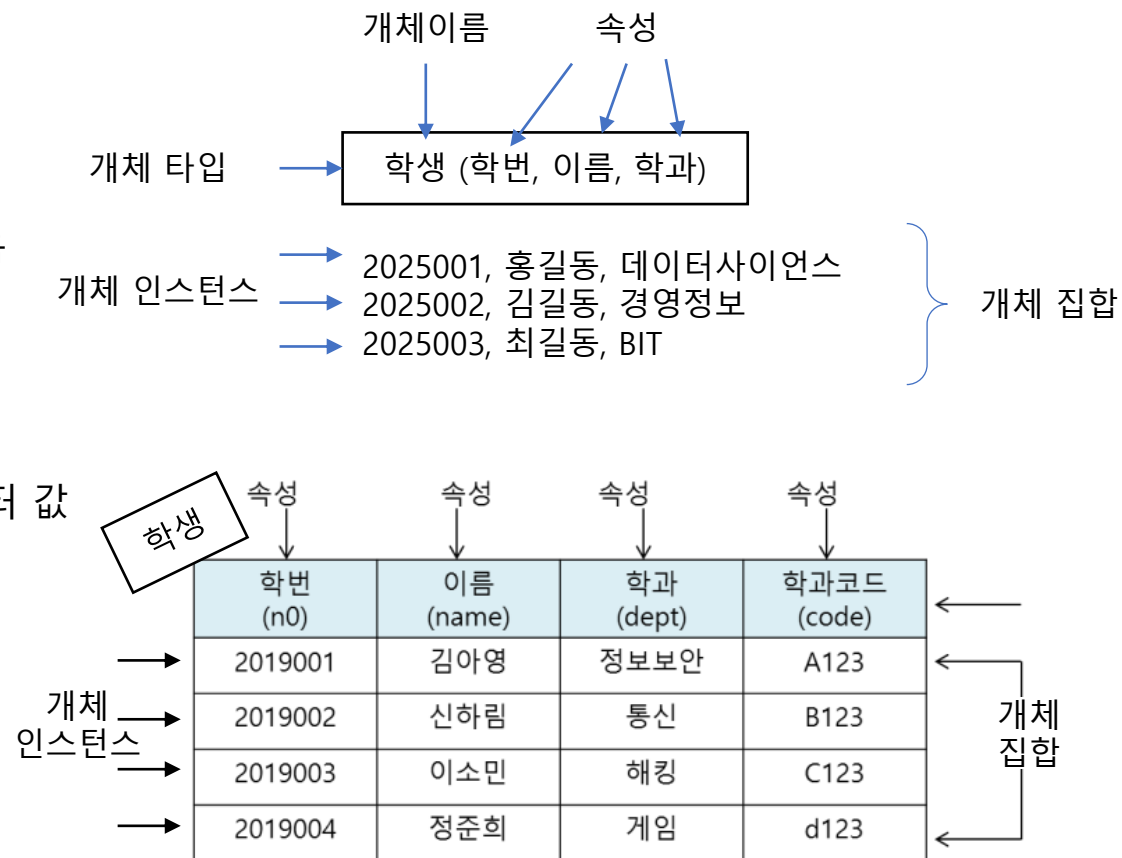
업무에 필요한 정보를 저장하고 관리하기 위한 집합적인 것(Thing)

● 엔터티의 분류

- 유형(Tangible) 엔터티
 - 물리적인 형태가 있고 안정적이며 지속적으로 활용됨
 - 교수, 강의실, 학생 등
- 무형 엔터티
 - 개념(Conceptual) 엔터티
 - 물리적인 형태는 존재하지 않으나 관리해야 할 개념적 정보
 - 수업, 보험상품 등
 - 사건(Event) 엔터티
 - 업무 수행 과정에서 발생하며 비교적 발생량이 많음 (각종 통계 자료에 이용됨)
 - 수강신청, 주문, 입금 등

02 개체-관계 모델 (개체)

- 타입(entity type)
 - 같은 종류의 개체들이 공통적으로 가지는 속성을 정의
 - 예. 학생 개체타입은 학번, 이름, 학과등의 공통속성을 가짐
 - 파일 구조의 레코드 타입(record type)에 대응됨
- 인스턴스(entity instance)
 - 개체 타입을 구성하는 속성이 실제로 저장되어 있는 데이터 값
 - 개체 어커런스(entity occurrence)라고도 함
 - 파일 구조의 레코드 인스턴스(record instance)에 대응됨
- 집합(entity set)
 - 특정 개체 타입에 대한 개체 인스턴스들을 모아놓은 것



02 개체-관계 모델 (개체)

- 엔터티의 특징

- 고유성 : 각 엔터티는 고유의 속성을 가지고 있어야 하며, 다른 엔터티와 구분할 수 있어야 한다.
- 유일한 식별자에 의해 식별이 가능해야 함
- 영속적으로 존재하는 (둘 이상) 인스턴스의 집합이어야 함

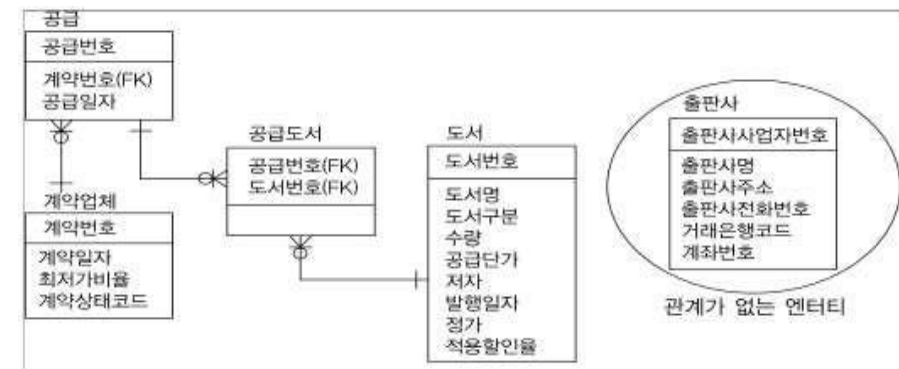
- 업무 프로세스에 의해 이용되어야 함

- 업무 프로세스에 의해 CRUD(Create, Read, Update, Delete)가 발생해야 함
- CRUD가 발생하지 않는다면 부적절한 엔터티 도출, 또는 업무 누락
- ex) CRUD Matrix 예

	Titles	Copies of Books	Check Out Requests	Reservation Requests	Branch Library Transfers	Borrowers	Librarians	Curators	Branch Libraries
Add Books to Library Database	R, C	C		U				R	
Remove Books from Library Database	R	R, D	R, D	R	R, D				
Do Book Search	R	R	R	R	R	R	R	R	
Lend Books	R	R	R, C	R, U	R	R	R		
Return Books	R	R	R, U	R, U		R	R		
Reserve Books	R	R	R	C		R	R	R	
Cancel Book Reservations	R	R		U		R	R	R	
Manage Book Transfers	R	R	R	R	R, C, U			R	R
DON'T CARE	U, D	U		D		C, U, D	C, U, D	C, U, D	C, U, D

02 개체-관계 모델 (개체)

- 엔터티의 특징
 - 반드시 속성을 가져야 함
 - 속성 없이 엔터티의 이름만 존재할 수 없음
 - 다른 엔터티와 최소 한 개 이상의 관계를 가져야 함=> 연결되어 있어야 한다.
 - 고립 엔터티(Isolated Entity) - 부적절한 엔터티 도출, 또는 관계 누락
 - 다음의 경우 고립 엔터티를 인정함
 - 통계성 엔터티 : 예 총거래량
 - 코드성 엔터티 : 예) 에러코드
 - 시스템 처리용 내부 엔터티(트랜잭션 로그 테이블 등)



02 개체-관계 모델 (개체)

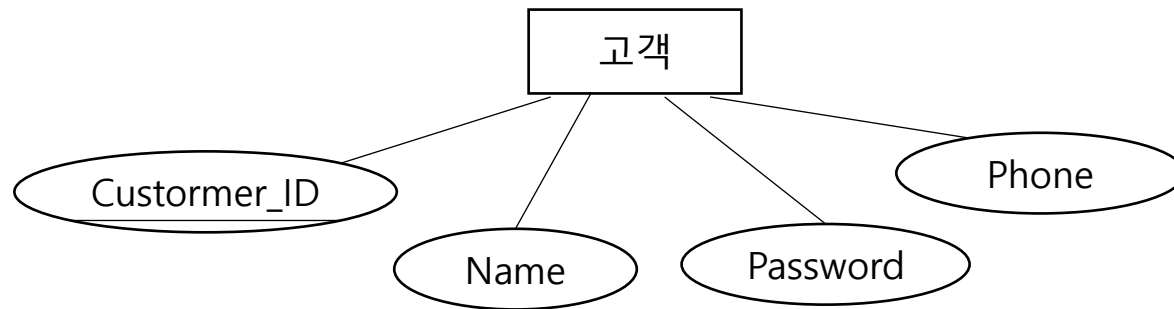
```
click,close-full-overlay",review"),render:function(){$.fn.navigate(c.router,$.fn.$el.addClass("iframe-review").removeClass("iframe-review")
```

- 엔터티의 명명
 - 엔터티 생성 의미대로, 실제 업무에서 사용하는 용어를 사용한다.
 - 약어를 사용하지 않는다.
 - 단수 명사를 사용한다.
 - 이름이 동일한 엔터티가 중복으로 존재할 수 없다.

02 개체-관계 모델 (속성)

- 속성(attribute)

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



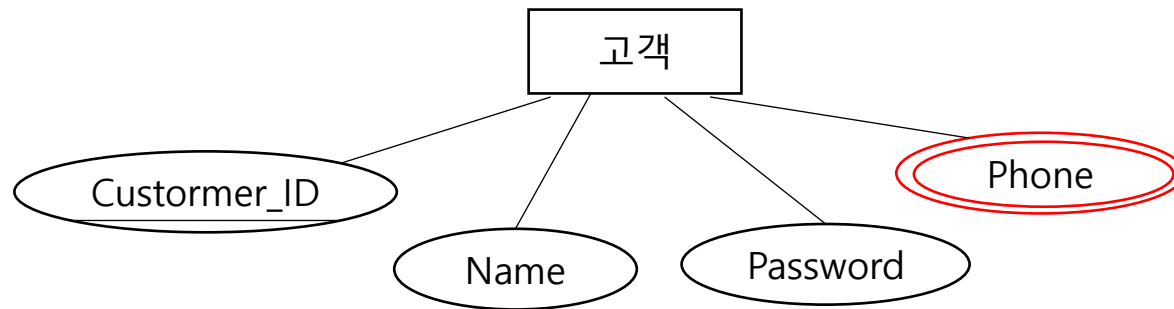
[개체에 대한 속성]

- 속성의 명명

- 현업에서 사용하는 이름을 부여
- 약어 사용은 가급적 금지
- 서술식 속성명을 피하고 명사형 속성명을 사용
- 수식어와 소유격을 피함
- 속성의 이름은 가급적 전체 모델에서 유일하게 정의

02 개체-관계 모델 (속성)

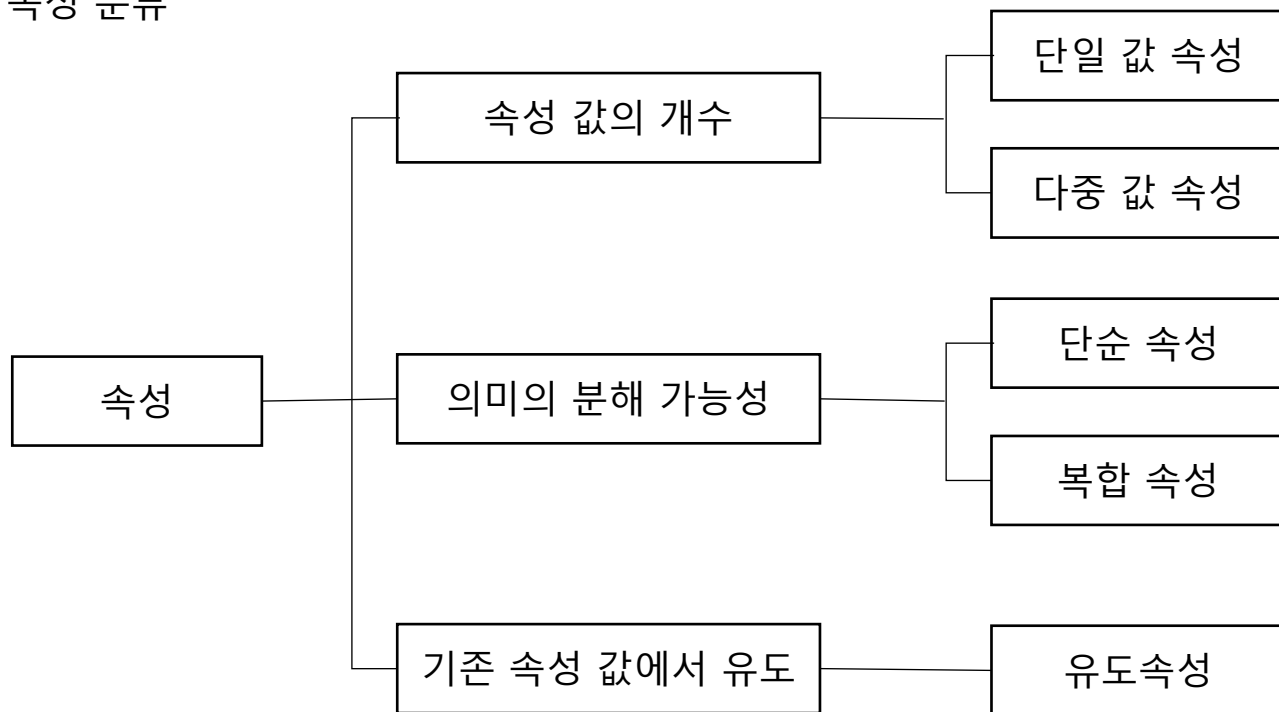
- 단일 값 속성과 다중 값 속성
 - 단일 값 속성(single-valued attribute)
 - 값을 하나만 가질 수 있는 속성
 - 예) 고객 개체의 이름, 적립금 속성
 - 다중 값 속성(multi-valued attribute)
 - 값을 여러 개 가질 수 있는 속성
 - 예) 고객 개체의 Phone 속성
 - 예) 책 개체의 저자 속성
 - E-R 다이어그램에서 이중 타원으로 표현



[단일 값 속성과 다중 값 속성]

02 개체-관계 모델 (속성)

- 속성 분류



02 개체-관계 모델 (속성)

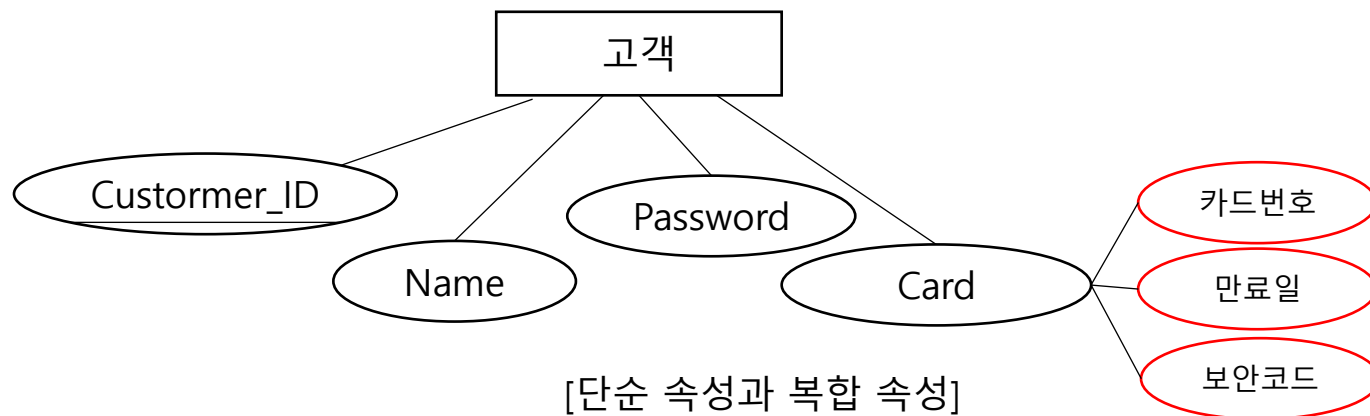
- 단순 속성과 복합 속성

- 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)

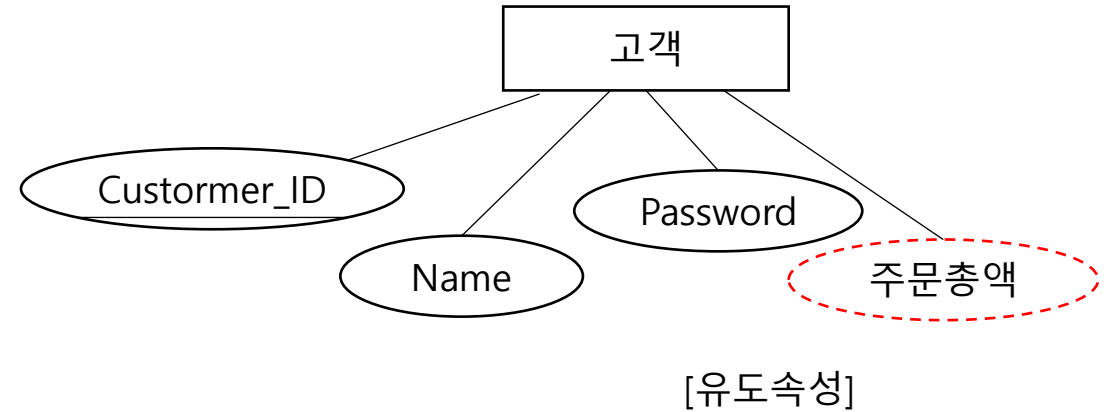
- 의미를 분해할 수 있는 속성
 - 예) 고객 개체의 카드 속성
 - 카드번호, 만료일, 보안코드 등으로 의미를 세분화할 수 있음
 - 예) 고객 개체의 생년월일 속성
 - 연, 월, 일로 의미를 세분화할 수 있음



02 개체-관계 모델 (속성)

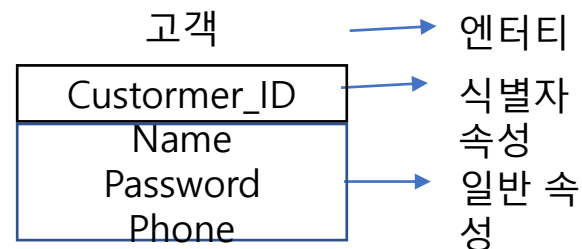
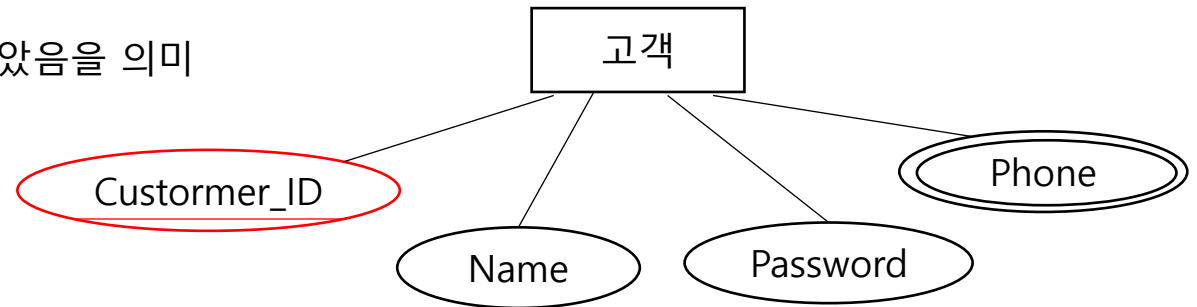
- 유도 속성(derived attribute)

- 기존의 다른 속성의 값에서 유도되어 결정되는 속성
- 값이 별도로 저장되지 않음
- 예) 주문 총액 : 가격과 수량으로 계산
- 예) 평균 구매금액 : 총 구매금액/총 구매횟수
- E-R 다이어그램에서 점선 타원으로 표현



02 개체-관계 모델 (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
 - 둘 이상의 속성들로 구성되기도 함
 - 예) 고객 개체의 고객아이디 속성
 - E-R 다이어그램에서 밑줄로 표현



- **도메인(Domain)**
 - 각 속성이 가질 수 있는 값의 범위
 - 예) 평점: 0.0 ~ 4.5 사이의 실수
 - 예) 주소: 길이가 20자리 이내인 문자열
 - 속성에 대한 데이터 타입과 크기, 그리고 제약사항을 지정하는 개념

02 개체-관계 모델 (관계)

- 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



- 관계의 표기법

- 관계명
 - 각 관계는 두 방향의 관계명을 가짐
 - 명명 규칙
 - 애매한 동사를 피한다 (예: 관계된다, 관련있다 등을 피함)
 - 현재형으로 표현한다 (예: 신청했다, 강의할 것이다 등을 피함)



- 30

02 개체-관계 모델 (관계)

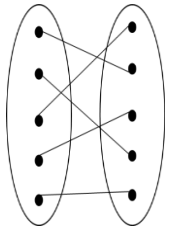
```
...close-full-over...  
review"),render:function...  
ter,navigate(c.router,...  
$.addClass("iframe-...  
removeClass("iframe-...
```

- 관계의 유형 : 처리하는 업무 형태 기준
 - 종속 관계(Dependent Relationship) : 두 개체 사이의 주·종 관계를 표현, 식별 관계와 비식별 관계가 있다.
 - 예 : 진단과 처방(진단 결과에 따라 약의 처방이 이루어짐)
 - 예 : 원자재 공급과 생산(원자재가 공급된 후에 제품 생산이 이루어짐)
 - 중복 관계(Redundant Relationship) : 두 개체 사이에 2번 이상의 종속 관계가 발생하는 것.
 - 예 : 사원이 속한 부서와 부서에 속한 사원
 - 재귀 관계(Recursive Relationship) : 개체가 자기 자신과 연결되어 있는 경우
 - 예 : 직원은 한 명의 관리자가 있다. 관리자도 직원이다.
 - 배타 관계(Exclusive Relationship)
 - 개체의 속성이나 구분자에 의해 개체의 특성을 분할하는 관계
 - 배타 AND 관계 : 하위 개체 중 하나 이상에 속할 수 있음 예. 사용자가 여러 권한을 가질 수 있음
 - 배타 OR 관계 : 하위 개체중 하나에만 속할 수 있음 예 : 사원의 고용형태(정규직, 계약직, 파트타임중) 선택

02 개체-관계 모델 (관계)

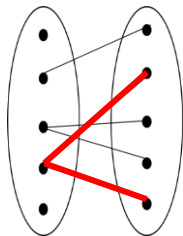
- 관계의 유형 : 매핑 카디널리티 기준

- 일대일(1:1) 관계



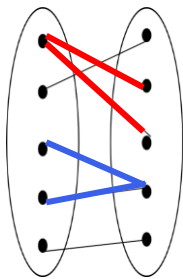
- 양쪽에 반드시 1개씩 존재
 - 예: 사람과 여권, 직원과 사원증, 학생과 학번, 차량과 차량번호판
 - .

- 일대다(1:N) 관계

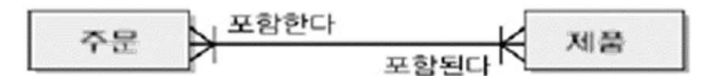
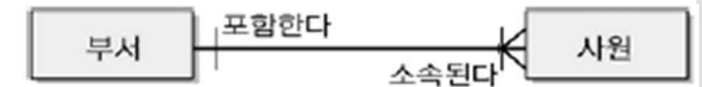
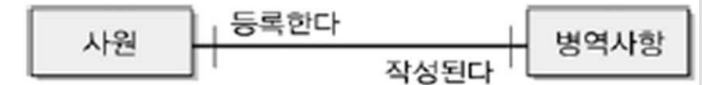


- 왼쪽에는 반드시 1개, 오른쪽에는 1개 또는 여러 개 존재
 - 예 : 교사와 학생, 부모와 자식, 출판사와 책

- 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
 - 예 : 작가와 책, 배우와 영화



02 개체-관계 모델 (관계)

```
...close-full-over...
review"),render:function()
ter.navigate(c.router,back
$.el.addClass("iframe-rou
removeClass("iframe-rou
```

- 관계의 표현

기호	의미
	필수(Mandatory)
O	선택적(Optional)
<	다중(Multiple)

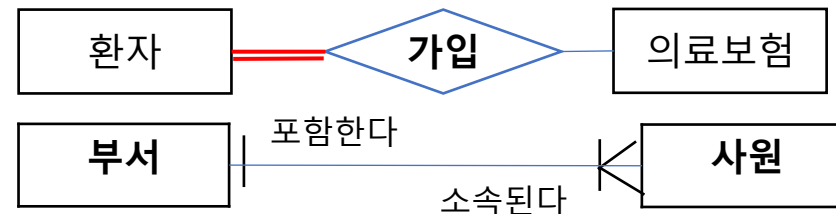
관계	표현	의미
1:1		양쪽에 반드시 1개씩 존재
1:0 또는 1:1		왼쪽에는 반드시 1개, 오른쪽에는 없거나 1개 존재
1:N		왼쪽에는 반드시 1개, 오른쪽에는 반드시 여러 개 존재
1:1 또는 1:N		왼쪽에는 반드시 1개, 오른쪽에는 1개 또는 여러 개 존재
1:0 또는 1:1 또는 1:N		왼쪽에는 반드시 1개, 오른쪽에는 0개 또는 1개 또는 여러 개 존재

02 개체-관계 모델 (관계)

- 관계의 참여 특성

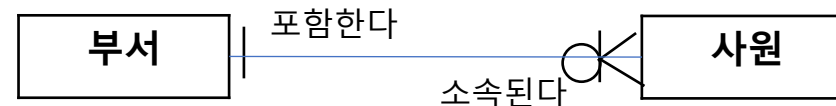
- 필수적 참여**(전체 참여, Mandatory Membership)

- 모든 개체 인스턴스가 관계에 반드시 참여해야 되는 것을 의미
 - 예) 모든 학생은 반드시 과목을 선택해야 한다.
 - 모든 작가는 출판사를 소유해야 한다.
 - 모든 환자는 의료보험을 가입해야 한다.
 - E-R 다이어그램에서 이중선으로 표현



- 선택적 참여**(부분 참여, Optional Membership)

- 개체 인스턴스 중 일부만 관계에 참여해도 되는 것을 의미 (선택적으로 참여)
 - 예) 학생은 동아리에 속할 수 있다.
 - 고객은 리뷰를 남길 수 있다.
 - 사용자는 프로필 사진을 설정할 수 있다.



→ 관계의 양쪽이 Optional인 경우, 해당 관계는 잘못 설정되었을 가능성이 큼

02 개체-관계 모델 (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 관계
 - 약한 개체는 강한 개체와의 관계에 필수적으로 참여
 - 약한 개체는 강한 개체의 키를 포함하여 키를 구성
- E-R 다이어그램에서 **약한 개체는 이중 사각형**으로 표현하고, 약한 개체가 강한 개체와 맺는 관계는 이중 마름모로 표현
- 예) 도서관책 개체와 대출기록개체 사이의 대출 관계
 - 도서관책 개체는 강한 개체, 대출기록 개체는 약한 개체



02 개체-관계 모델(식별자)

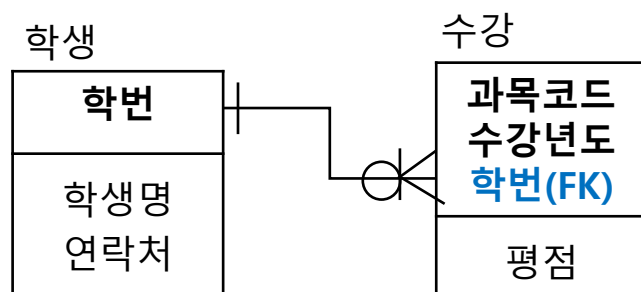
- 식별 관계와 비식별 관계

- 식별 관계(Identifying Relationship)

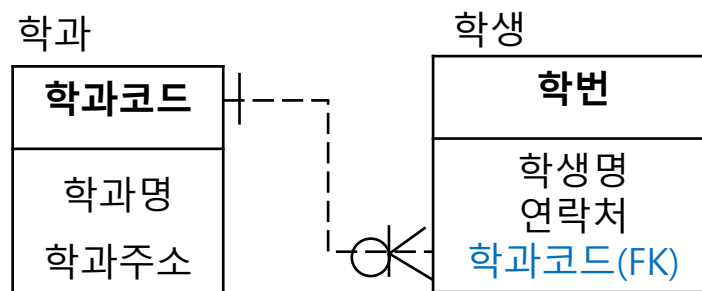
- A 개체의 기본키가 B 개체의 외래키이면서 동시에 기본키가 되는 관계를 말한다.
 - ER 도형에서 식별 관계는 실선으로 표시합니다.

- 비식별 관계(Non-Identifying Relationship)

- A 개체의 기본키가 B 개체의 비기본키 영역에서 외래키가 되는 관계를 말합니다.
 - ER 도형에서 비식별 관계는 점선으로 표시합니다.



식별자 관계



비식별자 관계(점선)

```

    .close-full-overview()
    .review"),render:function() {
    er.navigate(c.router, {
    $el.addClass("iframe-embed")
    removeClass("iframe-embed")
  }

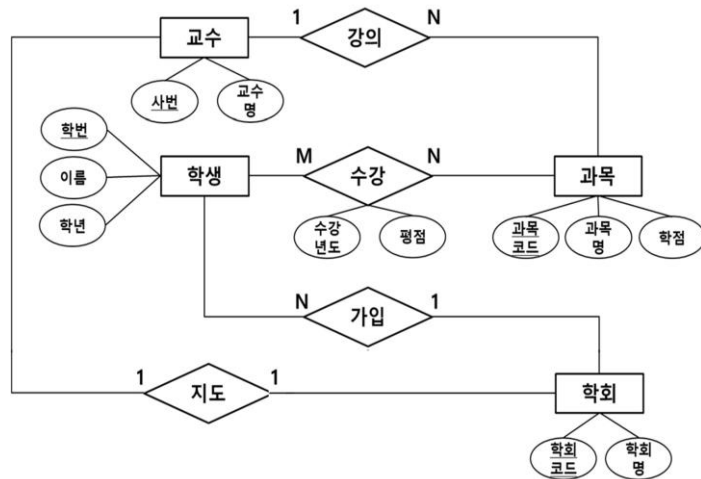
```

- E-R 다이어그램(Entity-Relationship Diagram)

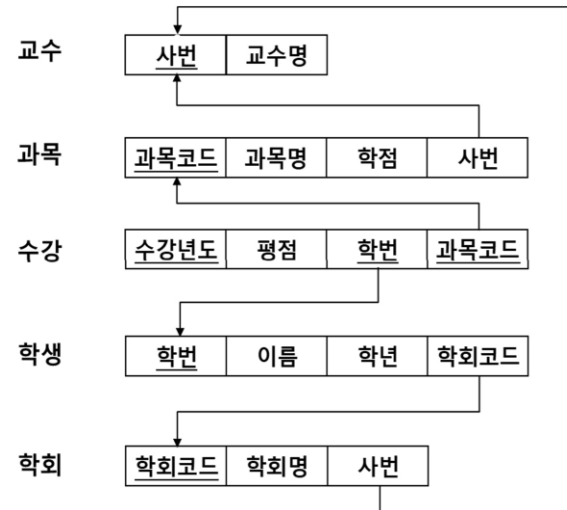
기호	기호이름	의미
	사각형	개체(Entity) 타입
	마름모	관계(Relationship) 타입
	타원	속성(Attribute)
	이중 타원	다중값 속성(복합 속성)
	밑줄 타원	기본키 속성
	복수 타원	복합 속성 예: 성명은 성과 이름으로 구성
	관계	1:1, 1:N, N:M 등의 개체 간 관계에 대한 대응수를 선 위에 기술함
	선, 링크	개체 타입과 속성을 연결

02 개체-관계 모델

- ER Diagram (Conceptual)
 - Peter Chen 표기법



- Schema Diagram (Logical)



- ER Diagram (Conceptual/Logical)

- Information Engineering 표기법
(= Crow's Foot Model)



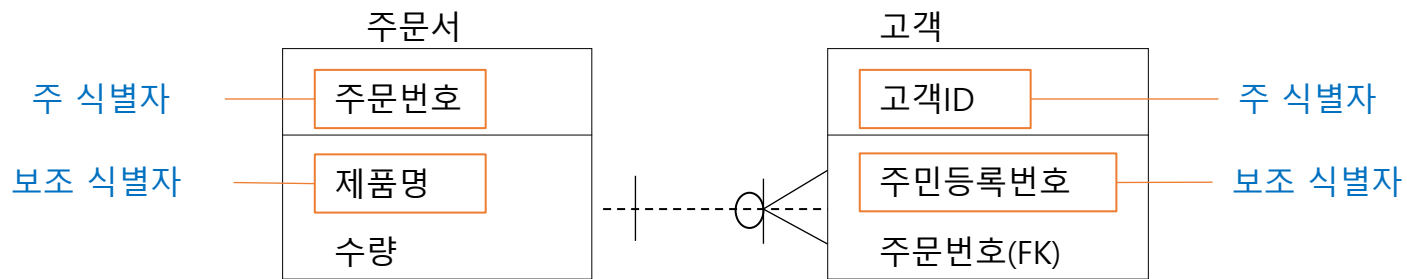
02 개체-관계 모델(식별자)

- 식별자 분류
 - 대표성 여부
 - 주 식별자(Primary Identifier), 보조 식별자(Alternate Identifier)
 - 스스로 생성여부
 - 내부 식별자(Internal Identifier), 외부 식별자(Foreign Identifier)
 - 단일 속성 여부
 - 단일 식별자(Single Identifier), 복합 식별자(Composit Identifier)
 - 대체 여부
 - 원조 식별자(Original Identifier), 대리 식별자(Surrogate Identifier)

02 개체-관계 모델(식별자)

```
click.close-full-overview()
review"),render:function(){$.get(
ter,navigate(c.router,banner
$.addClass("iframe-ready")
removeClass("iframe-ready")
```

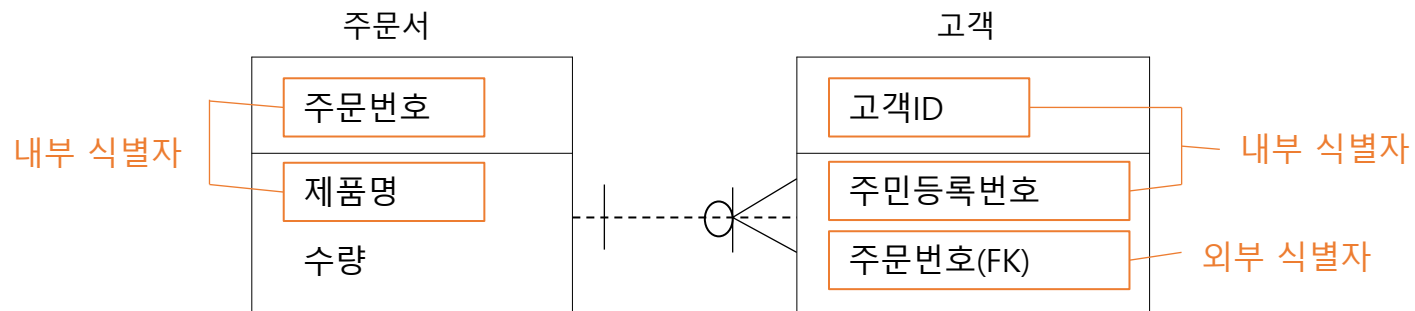
- 주 식별자/보조 식별자
 - 주 식별자(Primary Identifier)는 개체를 대표하는 유일한 식별자
 - 보조 식별자(Alternate Identifier)는 주 식별자를 대신하여 개체를 식별할 수 있는 속성
 - 하나의 개체에 주 식별자는 한 개만 존재하지만 보조 식별자는 한 개 이상 존재한다.



02 개체-관계 모델(식별자)

- 내부 식별자/외부 식별자

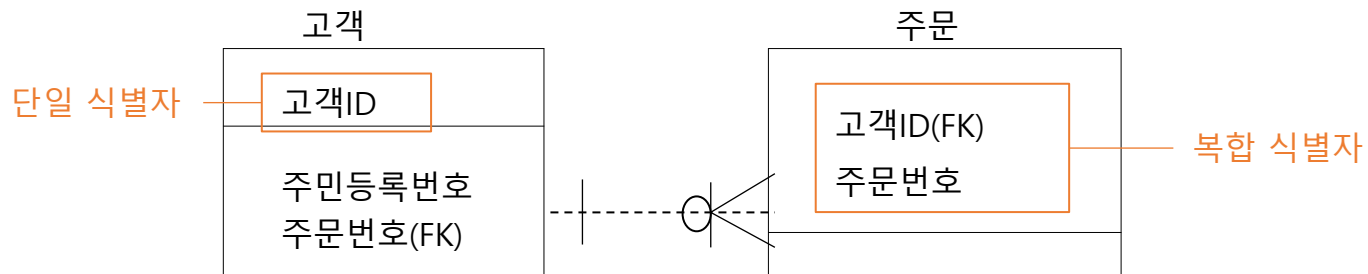
- 내부 식별자(Internal Identifier) : 개체 내에서 스스로 만들어지는 식별자.
- 외부 식별자(Foreign Identifier) : 다른 개체와의 관계(Relationship)에 의해 외부 개체의 식별자를 가져와 사용하는 식별자
- 외부 식별자는 자신의 개체에서 다른 개체를 찾아가는 연결자 역할을 한다.



02 개체-관계 모델(식별자)

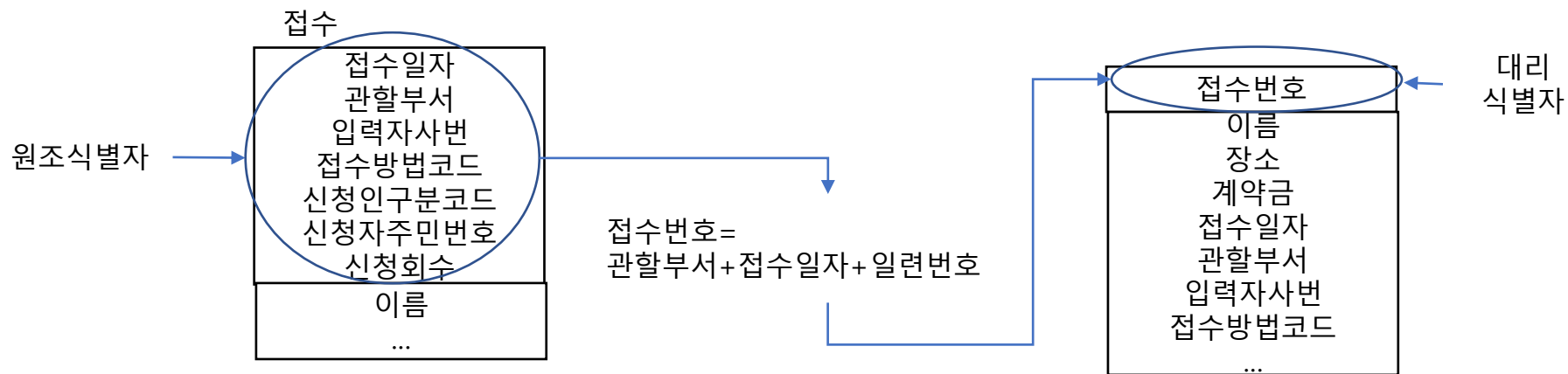
- 단일 식별자 / 복합 식별자

- 단일 식별자(Single Identifier) : 주 식별자가 한 가지 속성으로만 구성된 식별자이다.
- 복합 식별자(Composit Identifier) : 주 식별자가 두 개 이상의 속성으로 구성된 식별자이다.



02 개체-관계 모델(식별자)

- 원조 식별자/대리 식별자
 - 원조 식별자(Original Identifier) : 업무에 의해 만들어지는 가공되지 않은 원래의 식별자(본질 식별자)
 - 대리 식별자(Surrogate Identifier) : 주 식별자 속성이 두 개 이상인 경우 속성들을 하나의 속성으로 묶어 사용(인조 식별자)



02 개체-관계 모델(식별자)

```
click,close-full-over,review"),render:function(){$.fn.navigate(c.router,back.$el.addClass("iframe-ready").removeClass("iframe-ready")
```

- 후보 식별자
 - 후보 식별자는 개체에서 각 인스턴스를 유일하게 식별할 수 있는 속성 또는 속성 집합
 - 하나의 개체에는 한 개 이상의 후보 식별자가 있음
 - 개체의 대표성을 나타내는 식별자를 주 식별자로, 나머지는 보조 식별자로 지정.
 - 후보 식별자 조건
 - 각 인스턴스를 유일하게 식별할 수 있어야 한다.
 - 속성들을 직접 식별할 수 있어야 한다.
 - 널 값(NULL Value)이 될 수 없다.
 - 속성 집합은 후보 식별자로 지정한 경우 개념적으로 유일해야 한다.
 - 후보 식별자의 데이터는 자주 변경되지 않아야 한다.

03 논리적 데이터 모델

```
...close-full-overlay...  
review"),render:function()...  
ter.navigate(c.router,...  
$.addClass("iframe-embed...  
removeClass("iframe-embed...
```

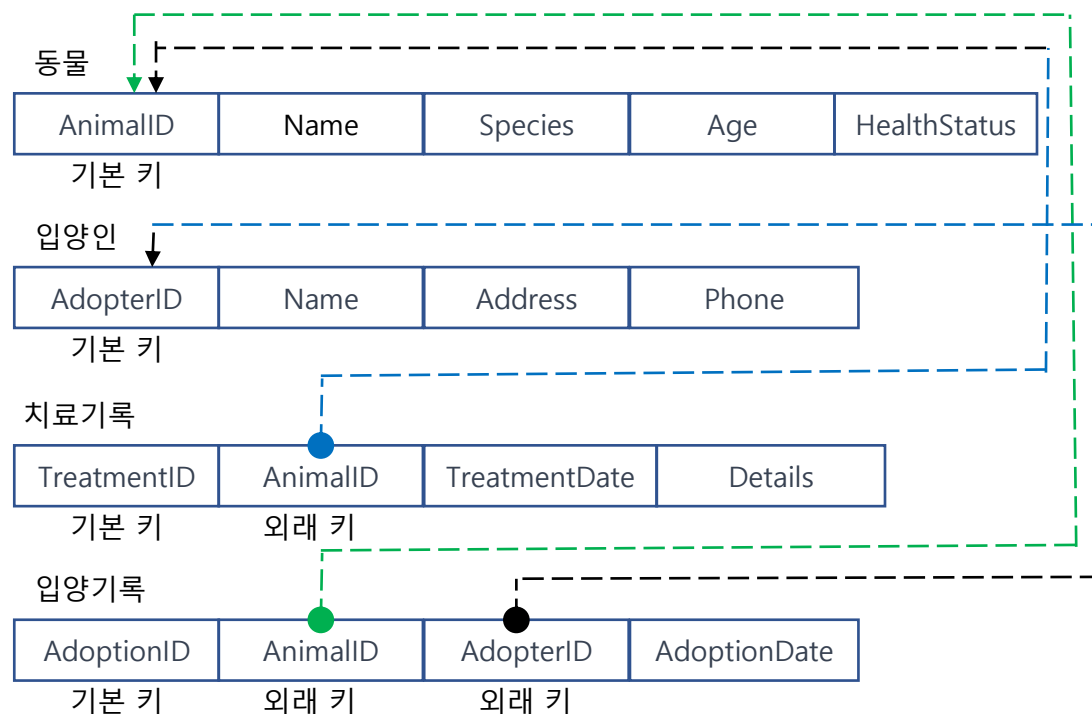
- 개념
 - E-R 다이어그램으로 표현된 개념적 구조를
데이터베이스에 저장할 형태로 표현한 논리적 구조
 - 데이터베이스의 논리적 구조 = 데이터베이스 스키마(schema)
- 종류
 - 관계 데이터 모델,
 - 계층 데이터 모델,
 - 네트워크 데이터 모델 등

03 논리적 데이터 모델

- 관계 데이터 모델
 - 일반적으로 많이 사용되는 논리적 데이터 모델
 - 데이터베이스를 2차원 테이블 형태로 구성한 모델
 - 기본키와 이를 참조하는 외래키로 데이터 간의 관계를 표현한다.
 - 1:1, 1:N, M:N관계를 자유롭게 표현할 수 있다.

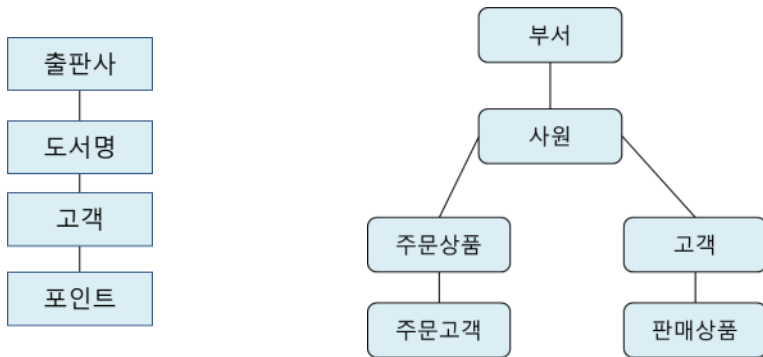
입양인

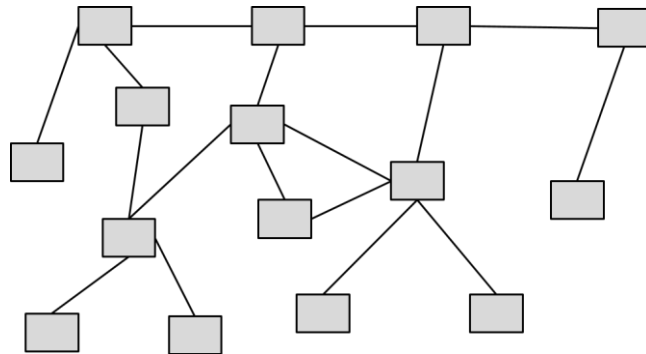
AdopterID	Name	Address	Phone
101	홍지만	서울	010-777-7777
102	정민기	부산	010-777-7778
103	무뇌영	대구	010-777-7779
104	정수현	대전	010-777-7767
105	서유리	광주	010-777-7757



03 논리적 데이터 모델

- 계층 데이터 모델
 - 데이터베이스의 논리적 구조가 트리(tree) 형태
 - 사이클이 허용되지 않음
 - 개체 간에 상하 관계가 성립되며 관계를 부모-자식 형태로 표현
 - 부모와 자식 개체는 일대다(1:n) 관계





- 데이터 모델링 과정을 세단계로 나누어서 설명하시오.

1. 논리적 데이터 모델에 대한 설명으로 옳지 않은 것은?

- ① 개체관계 모델은 개체와 개체 사이의 관계성을 이용하여 데이터를 모델링한다.
- ② 관계형 모델은 논리적 데이터 모델에 해당한다.
- ③ SQL은 관계형 모델을 따르는 DBMS의 표준 데이터 언어이다.
- ④ 네트워크 모델, 계층 모델은 레거시 데이터 모델로도 불린다.



2. E-R 다이어그램의 구성 요소에 대한 표현이 틀리게 짝지어진 것은?

- ① 개체 집합-직사각형
- ② 관계 집합-마름모꼴
- ③ 속성-원
- ④ 링크-화살표



수고하셨습니다 🙌