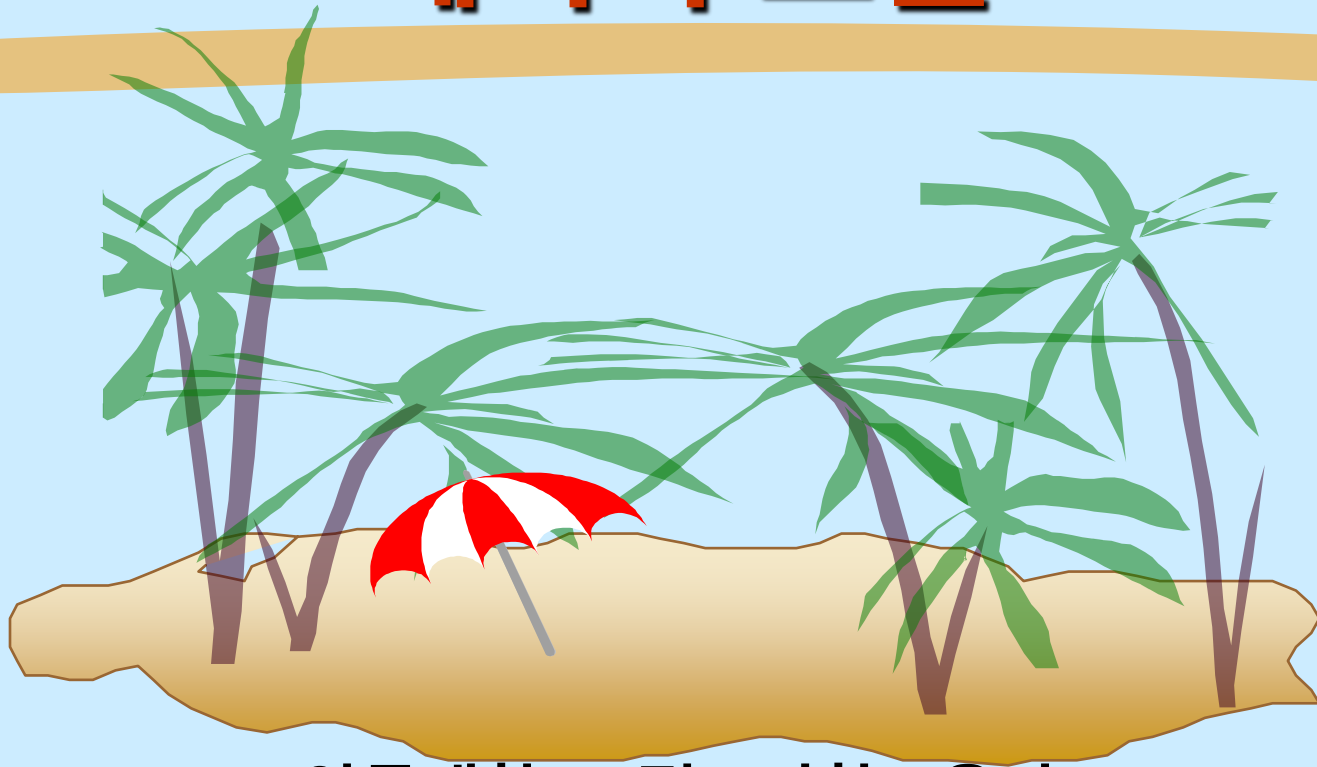
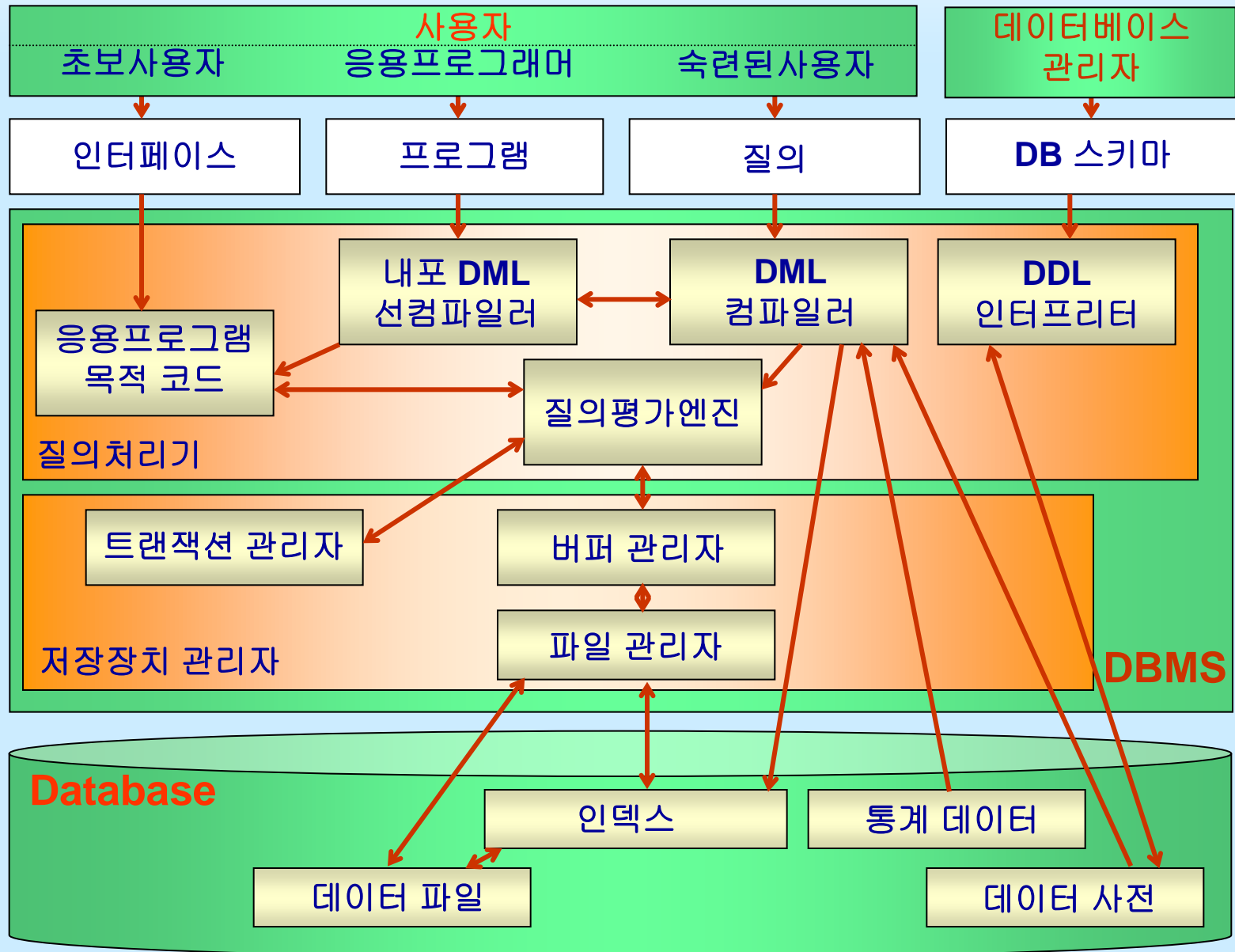


데이터 모델



안동대학교 정보과학교육과

데이터베이스시스템 구성도





□ 데이터 모델의 개념

□ 개체-관계 데이터 모델

- 개체 / 관계 / 제약조건

- 개체-관계 모델/다이어그램

□ 관계형 데이터 모델

- 관계형 데이터 구조

□ 개체-관계 모델을 관계 모델로 변환

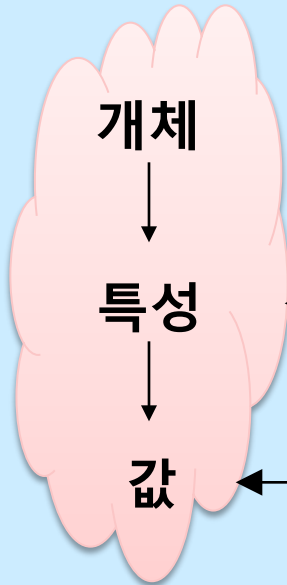
컴퓨터가 관리하는 데이터베이스는 계속적으로 변화하는 현실 세계를 표현

컴퓨터에 저장할 데이터의 구조를
논리적으로 표현하기 위해 사용하는 도구

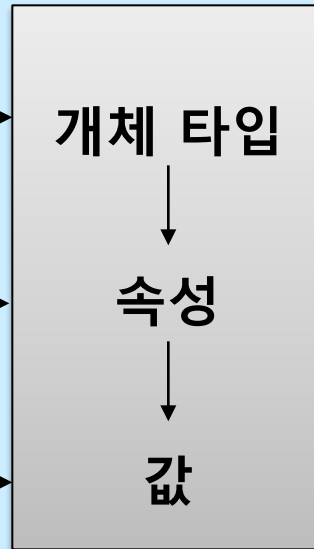
데이터 모델

데이터 : 현실세계 vs. 컴퓨터

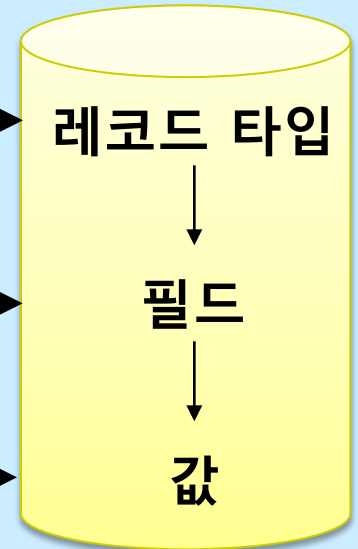
현실 세계(개체)



개념 세계(개념)



컴퓨터 세계(데이터)



값
사실

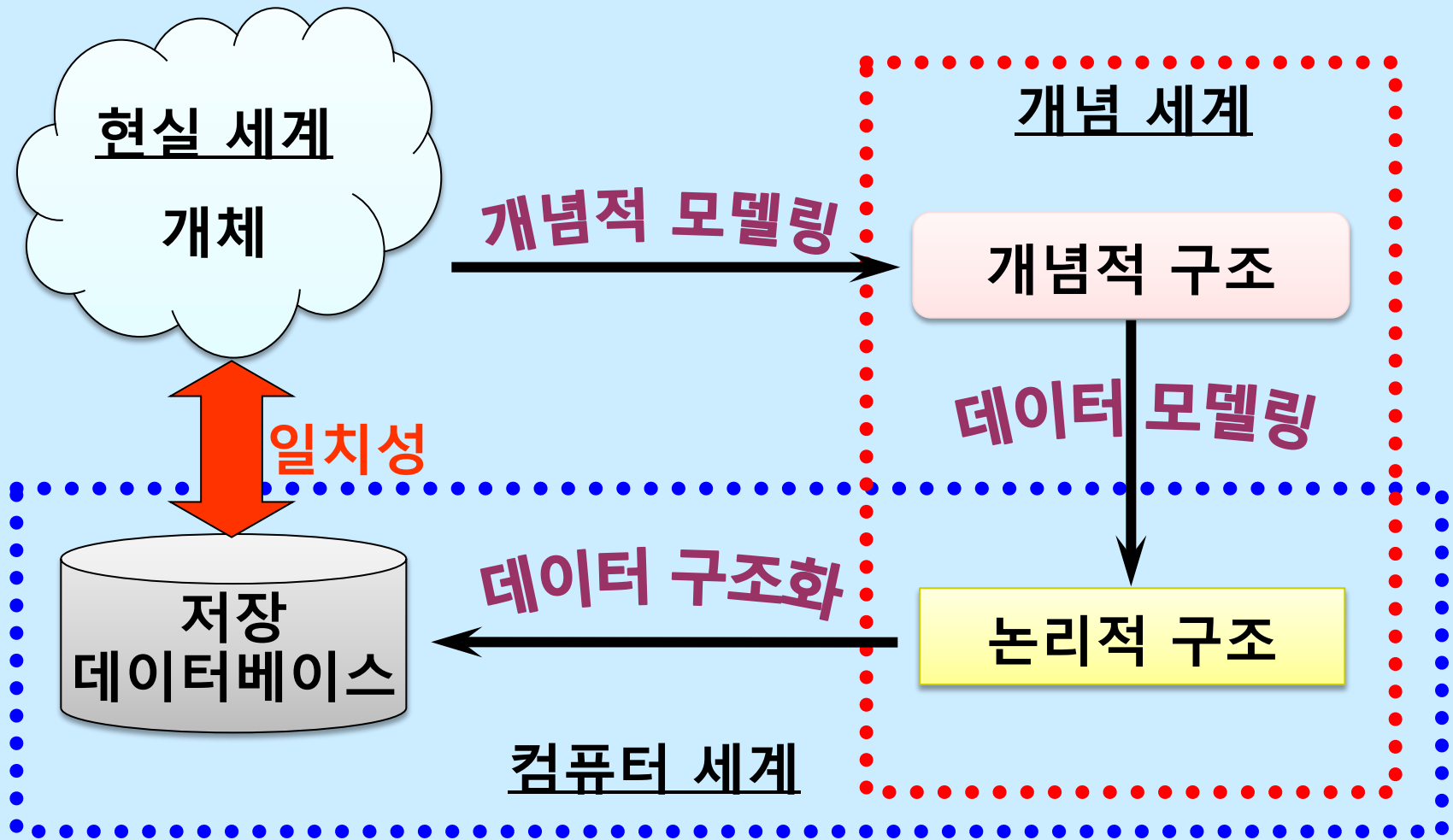
추상화
개념적
모델링

추상적
개념적 표현

변환
데이터
모델링

데이터 구조의
논리적 표현

데이터베이스 : 현실세계 vs. 컴퓨터



데이터 모델(Data Model) - I

□ 다음 사항들을 기술하기 위한 개념적 표현

- 데이터
- 데이터들 간의 관계
- 데이터의 의미
- 일관성 제약 조건

□ 데이터 모델 : $D = \langle S, O, C \rangle$

□ S : 데이터의 구조(Structure)

- 정적 성질 (추상적 개념) - 개체타입과 이들 간의 관계를 명세

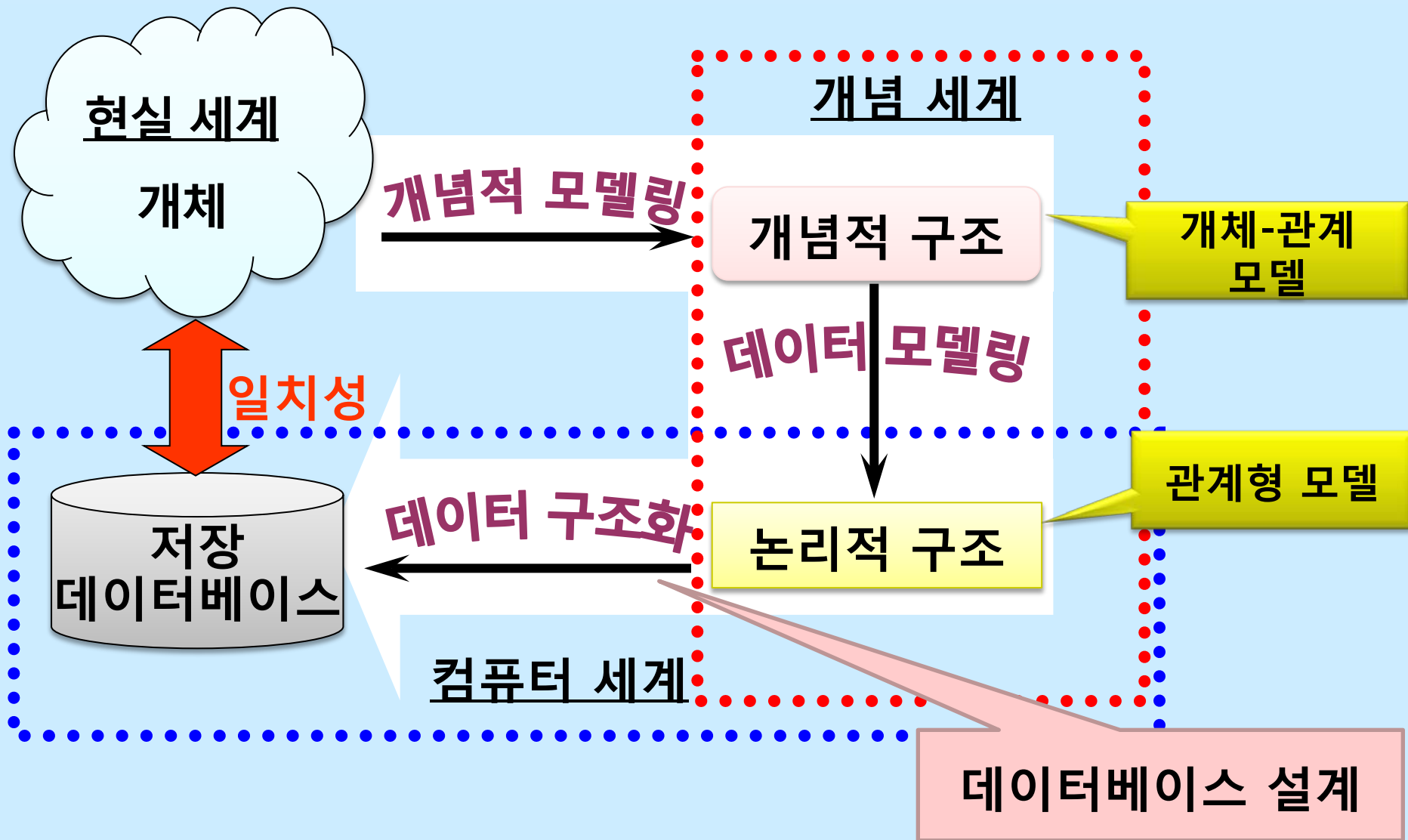
□ O : 연산(Operations)

- 동적 성질 - 개체 인스턴스를 처리하는 작업(데이터 조작)에 대한 명세

□ C : 제약 조건(Constraints)

- 데이터의 논리적 제약 - 개체 인스턴스의 허용 조건
- 데이터 조작의 한계를 표현한 규정

데이터 모델(Data Model) - II



배울 내용

□ 데이터 모델의 개념



□ 개체-관계 데이터 모델

□ 개체 / 관계 / 제약조건

□ 개체-관계 모델/다이어그램

□ 관계형 데이터 모델

□ 관계형 데이터 구조

□ 개체-관계 모델을 관계 모델로 변환

개체

□ 개체(Entity)

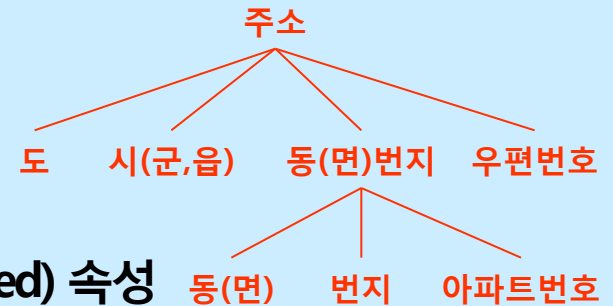
- 독립적으로 존재하며 다른 모든 개체와 구별되는 객체

□ 개체 타입(Entity Type)

- 이름과 속성들(attributes)로 정의

□ 속성의 유형

- 단순(simple) 속성과 복합(composite) 속성
- 단일값(single-valued) 속성과 다중값(multi-valued) 속성
- 저장된(stored) 속성과 유도된(derived) 속성
- 널(null) 속성



□ 개체 집합(Entity Set)

- 특정 개체 타입에 대한 인스턴스 집합

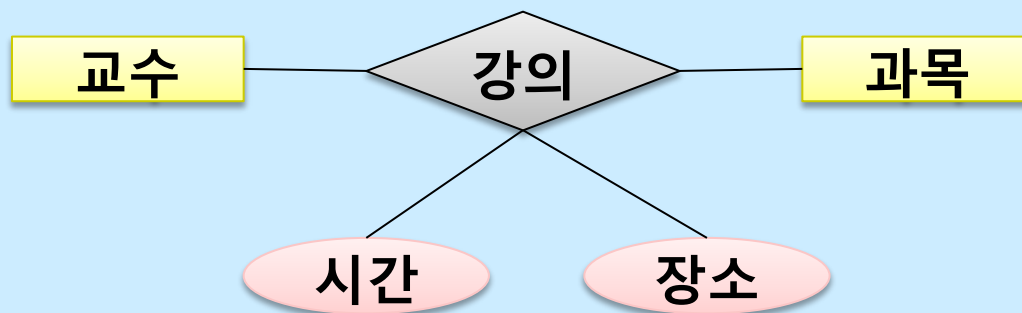
관계

□ 관계(Relationship)

- 여러 개체들 사이의 연관성

□ 관계 타입(Relationship type)

- 개체 집합들 간의 대응성(correspondence), 사상(mapping)
- 설명 속성(descriptive attribute)을 가질 수 있음



□ 관계 집합(Relationship Set)

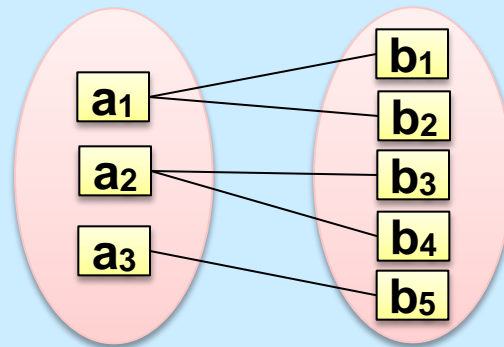
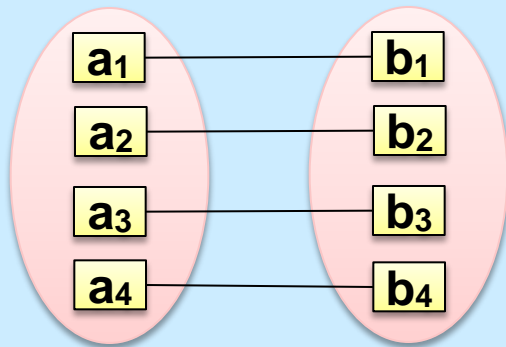
- 개체 집합 간에 실제로 이루어진 관계 인스턴스 집합
- $\{ (e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n \}$,
여기서 $(e_1, e_2, \dots, e_n) = \text{relationship}$

제약조건(I)

□ 관계 타입의 유형 - 대응수(mapping cardinality)

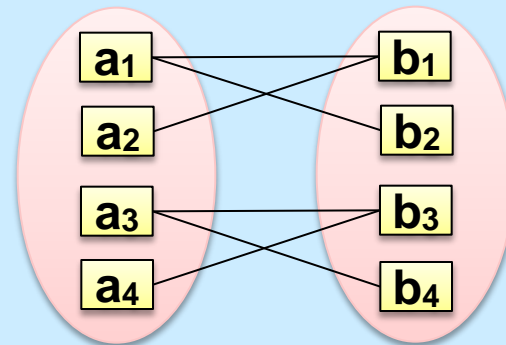
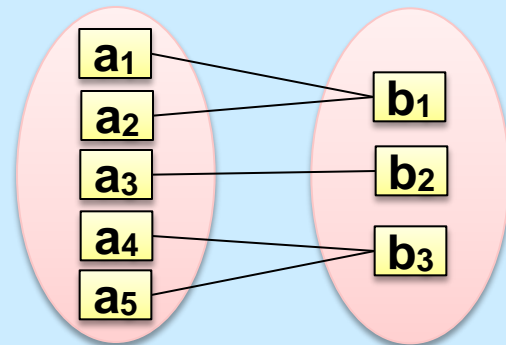
□ 1:1 (일 대 일) $f_x : x \rightarrow y$ and $f_y : y \rightarrow x$

□ 1:n (일 대 다) **not** $f_x : x \rightarrow y$ **but** $f_y : y \rightarrow x$



□ n:1 (다 대 일) $f_x : x \rightarrow y$ **not** $f_y : y \rightarrow x$

□ n:m (다 대 다) **not** $f_x : x \rightarrow y$ **and** **not** $f_y : y \rightarrow x$



제약조건(II)

□ 관계 타입의 특성 - 참가 제약조건

□ 전체 참여(total participation)

- A-B 관계에서 개체 집합 B의 모든 개체가 A-B 관계에 참여
- 예: 학과-교수

□ 부분 참여(partial participation)

- A-B 관계에서 개체 집합 B의 일부 개체만 A-B 관계에 참여
- 예 : 과목-학생(휴학생 허용 시)

□ 존재 종속(existence dependency)

- 어떤 개체 b의 존재가 개체 a의 존재에 좌우됨
- b는 a에 존재 종속
- a: 주 개체(dominant entity), b: 종속 개체(subordinate entity)

제약조건(III)

개체(관계) 집합에서 특정 개체(관계)를 구별하는 방법 → 키(key)

키 지정 : 모델링하는 실세계의 제약 조건중의 하나임

□ 개체 집합에서의 키

□ 수퍼 키(super key)

- 개체 집합에서 한 개체를 유일하게 식별할 수 있도록 해 주는 하나 이상의 속성들의 집합

□ 후보 키(candidate key)

- 최소 길이로 이루어진 수퍼 키

□ 주 키(primary key)

- 여러 개의 후보 키 중에서 데이터베이스 설계자에 의해 선택된 후보 키

□ 관계 집합에서의 키

□ 후보 키(candidate key)

- 관계에 참여하는 각 엔티티 집합의 주 키의 결합

□ 주 키(primary key)

- 대응 수와 관계 집합의 의미를 고려하여 결정해야 함

배울 내용

□ 데이터 모델의 개념

□ 개체-관계 데이터 모델

□ 개체 / 관계 / 제약조건

□ 개체-관계 모델/다이아그램



□ 관계형 데이터 모델

□ 관계형 데이터 구조

□ 개체-관계 모델을 관계 모델로 변환

개체-관계 모델(E-R Model)

- 1976년 Peter Chen이 제안
- 개체 타입과 이들 간의 관계 타입을 이용해서 현실 세계를 개념적으로 표현하는 방법 → 개념적 데이터 모델
 - 개체 집합: 한 개체 타입에 속하는 모든 개체 인스턴스
 - 관계 집합: 한 관계 타입에 속하는 모든 관계 인스턴스
- E-R 다이어그램
 - 개체-관계 모델을 그래프 방식으로 표현한 것
- 개념적 모델링
 - 인간의 이해를 위해 현실 세계에 대한 인식을 추상적 개념으로 표현하는 과정
 - 결과: E-R Diagram

E-R 다이어그램

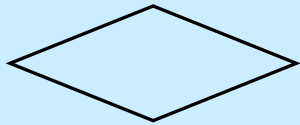
□ notation



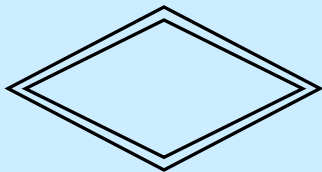
개체 타입



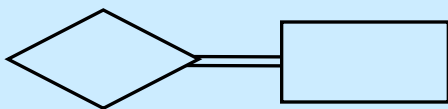
약한 개체 타입



관계 타입



식별 관계 타입



전체 참여 개체 타입



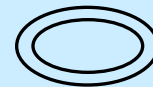
애트리뷰트



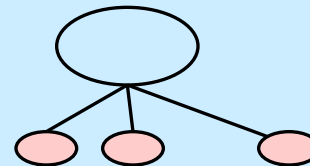
키 애트리뷰트



부분키 애트리뷰트



다중값 애트리뷰트

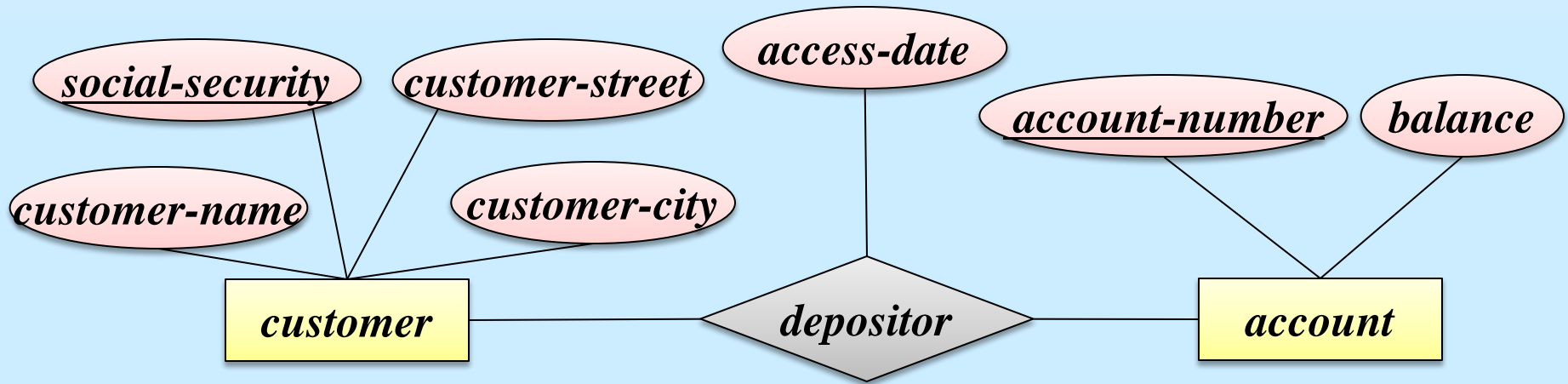


복합 애트리뷰트



유도 애트리뷰트

E-R 다이어그램의 예(I)



□ 은행업무

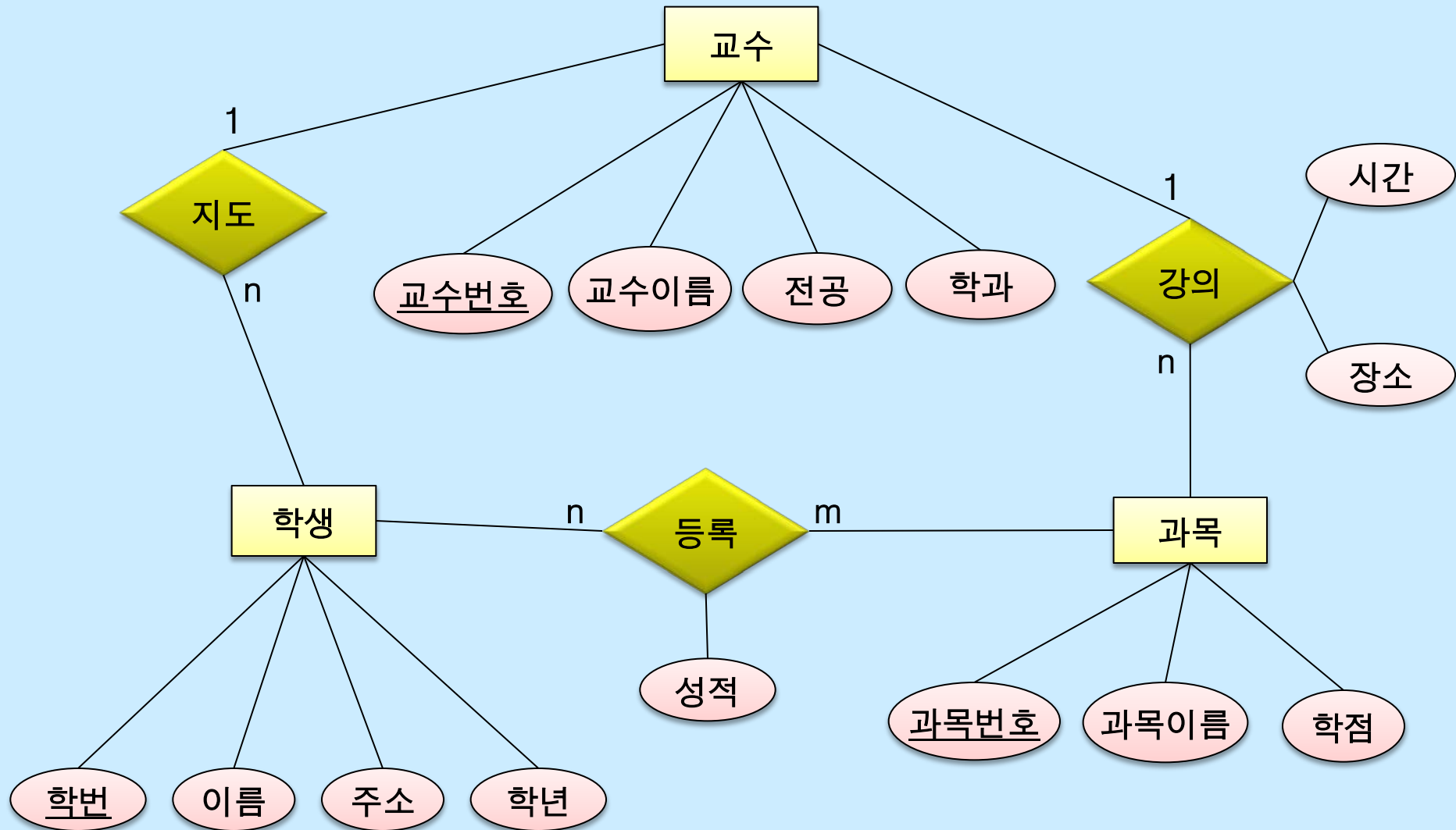
□ 고객(customer)과 고객의 계좌(account)에 대한 정보 관리

□ 구성요소

- 사각형: 개체 집합(고객, 계좌)을 나타냄
- 타원: 개체 집합, 또는 관계(예: deposit)를 특징짓는 속성을 나타냄
- 마름모: 개체 집합 간의 관계를 나타냄
- 선: 속성과 개체 집합, 개체 집합과 관계 사이를 연결

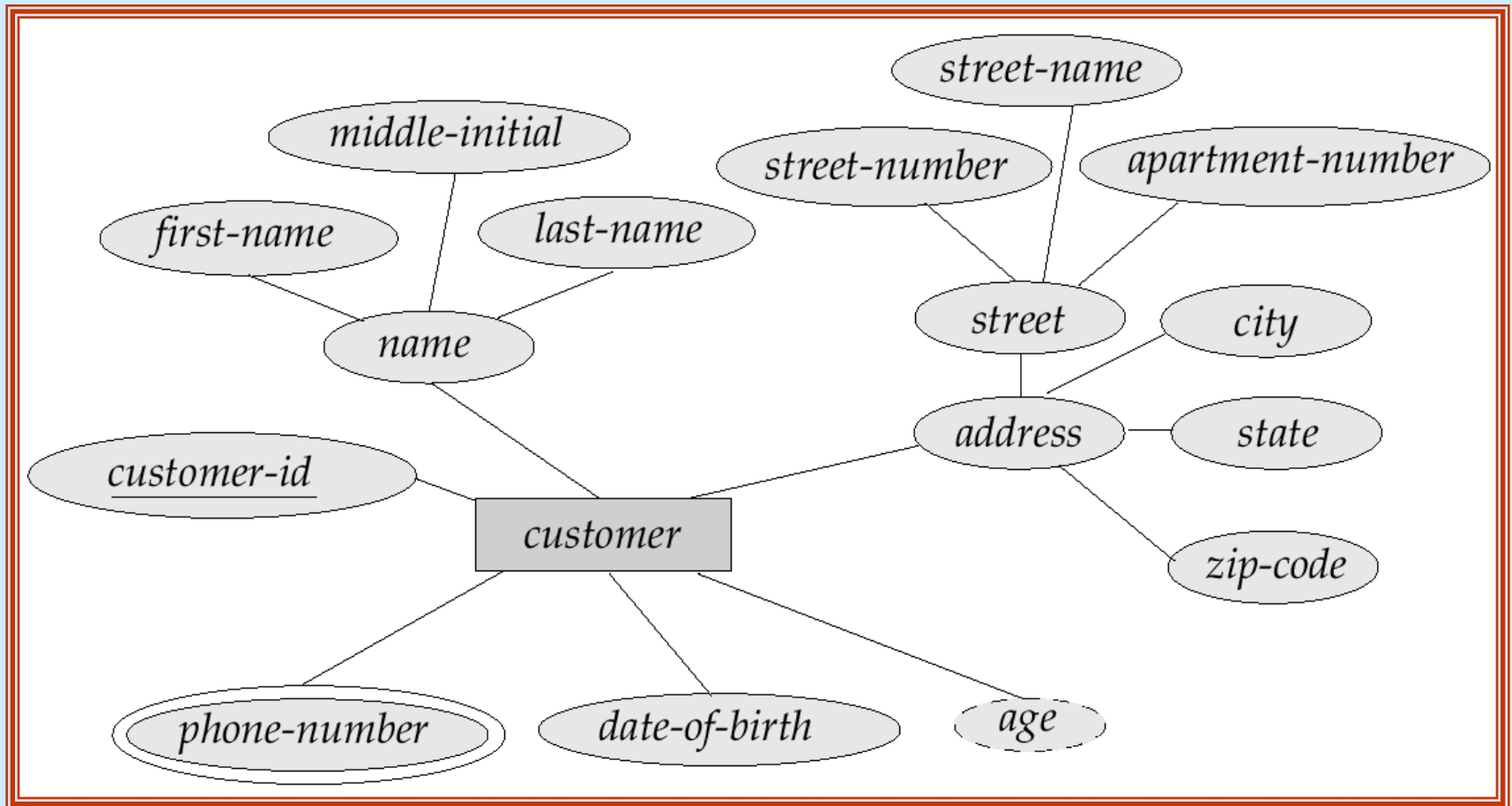
E-R 다이어그램의 예(II)

□ 학사업무



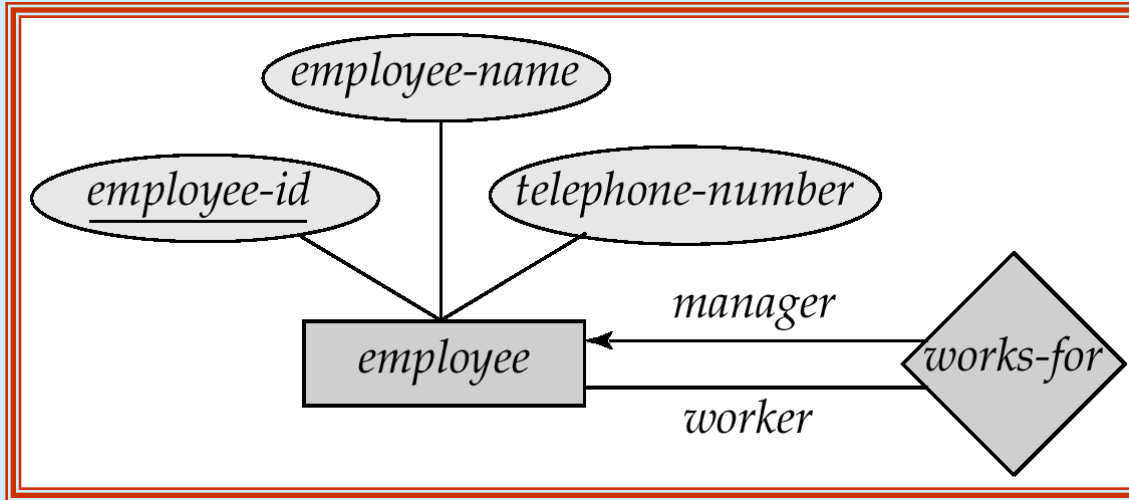
E-R 다이어그램의 예(III)

□ 복합, 다중, 유도 속성으로 구성된 E-R Diagram

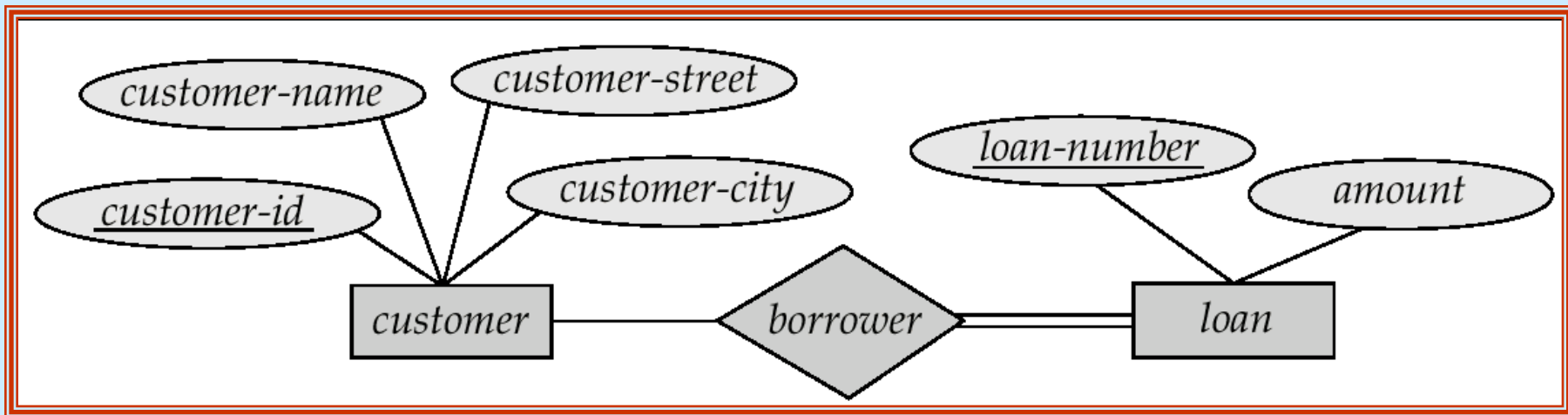


E-R 다이어그램의 예(IV)

□ 역할 표시자가 있는 E-R Diagram

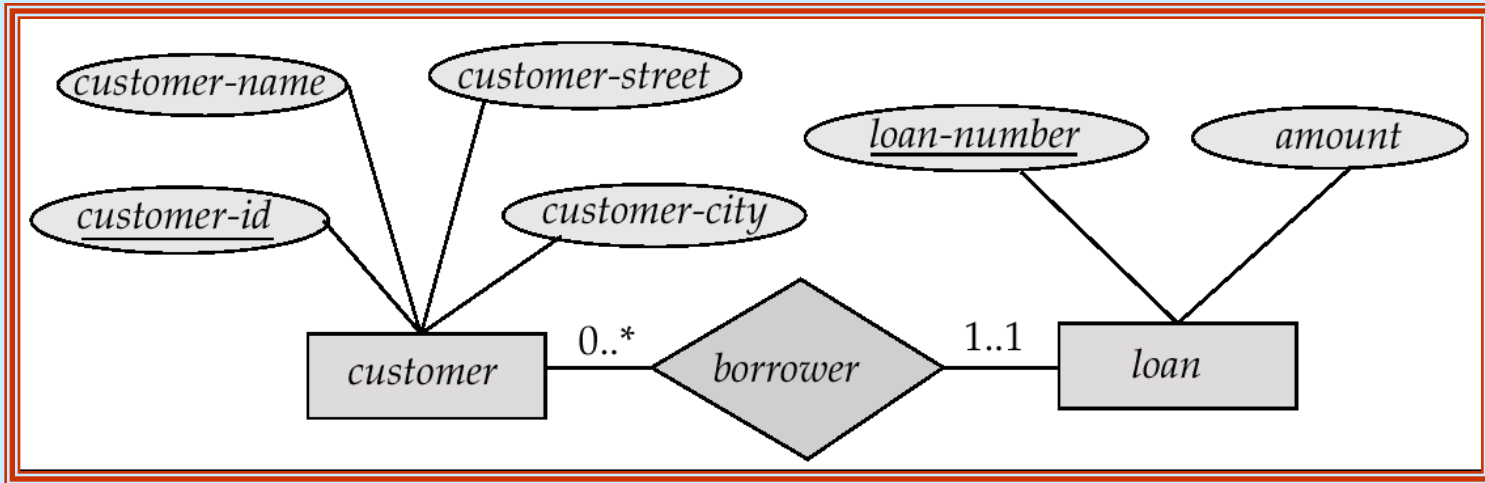


□ 전체적 참가

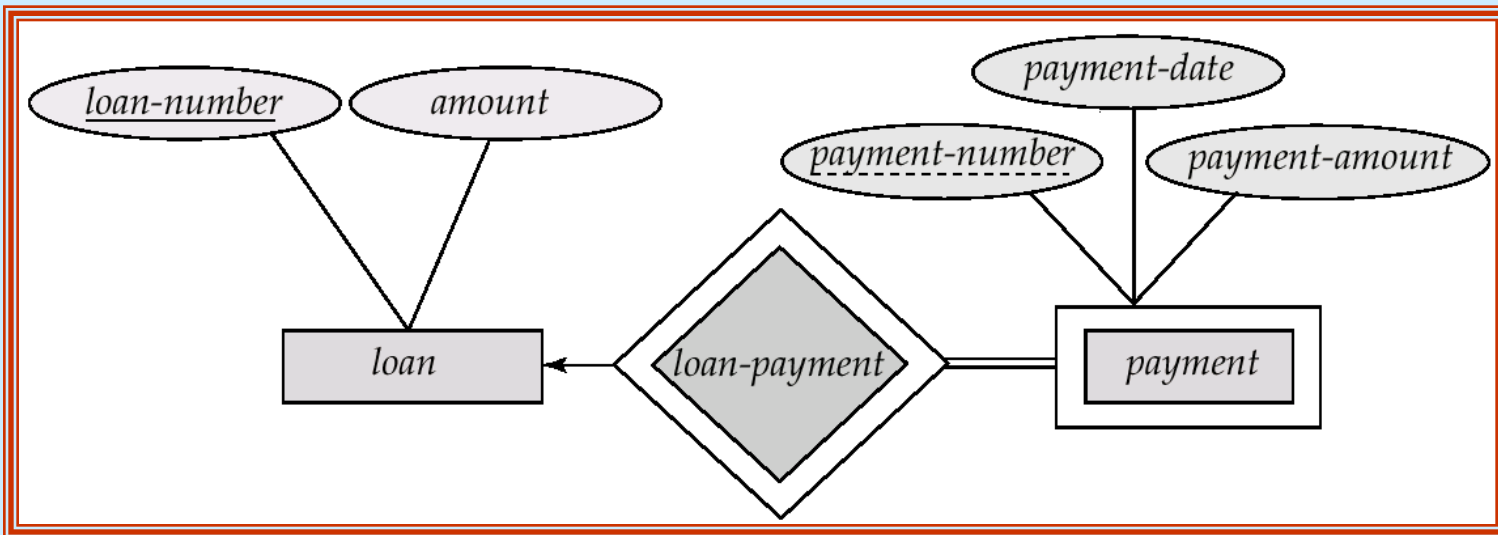


E-R 다이어그램의 예(V)

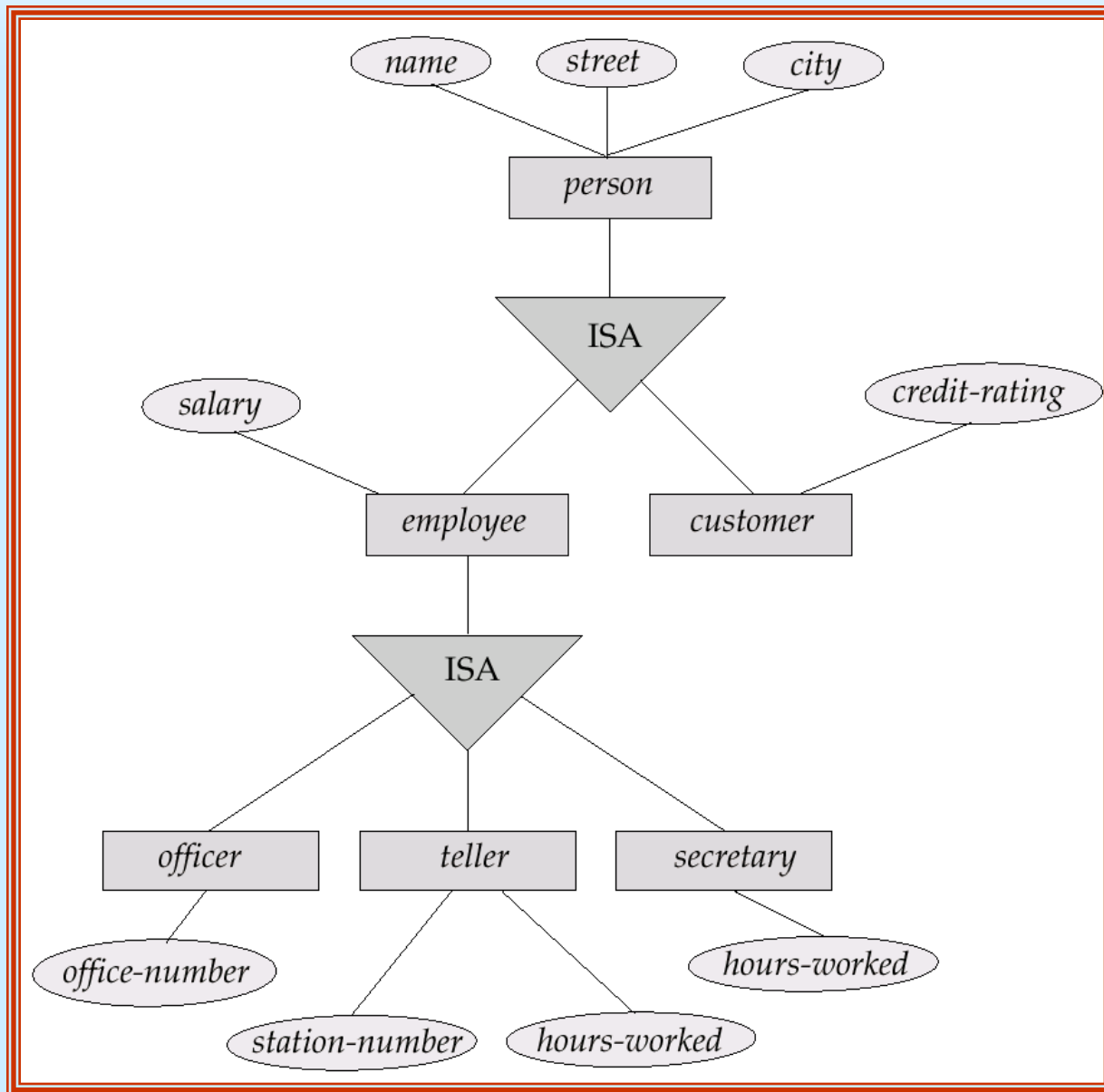
□ 관계 집합에서의 대응수 제약조건



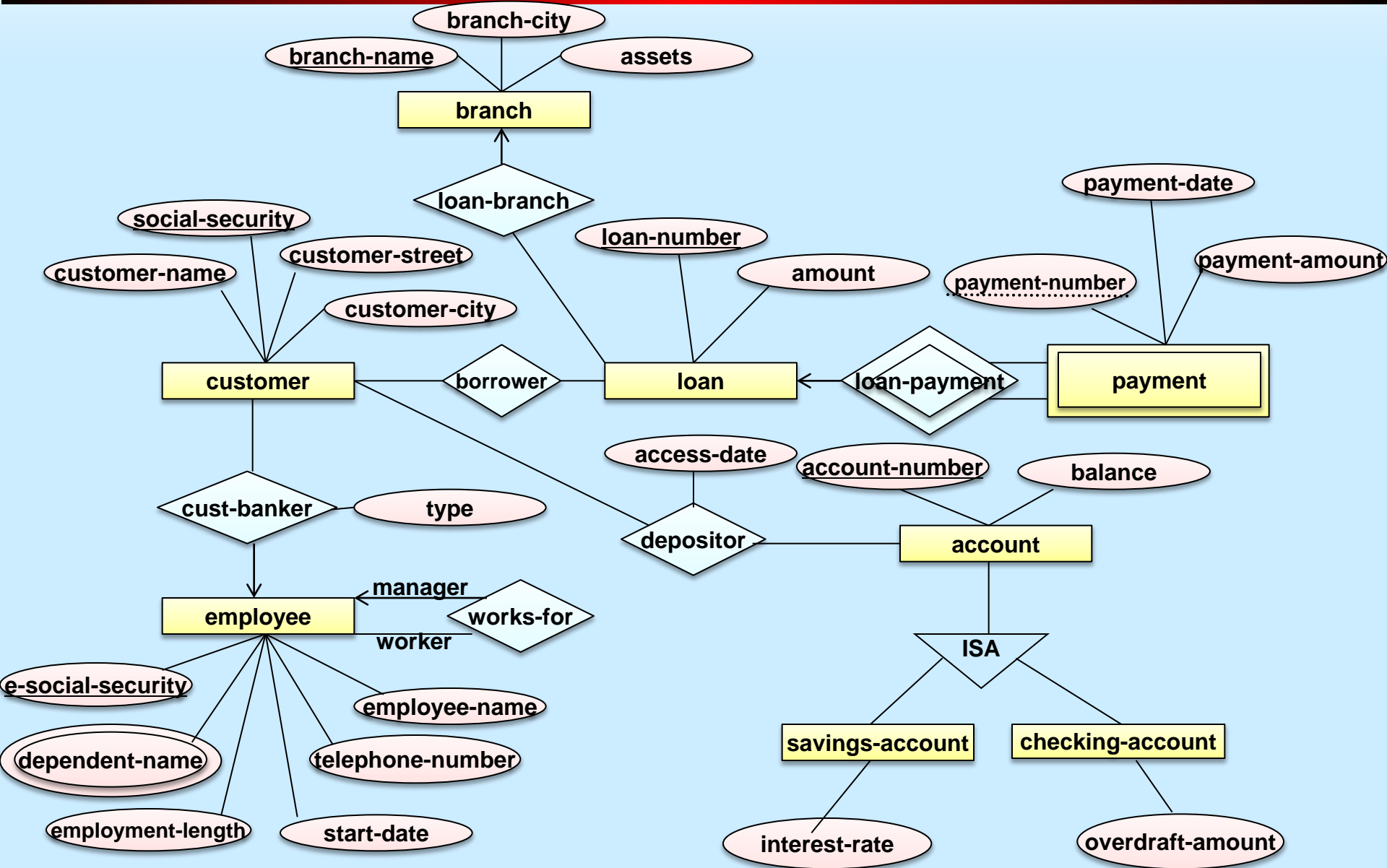
□ 약 개체 집합



E-R 모델의 확장 – 세분화/일반화



E-R 다이어그램의 예 - 은행기관



배울 내용

- 데이터 모델의 개념

- 개체-관계 데이터 모델

 - 개체 / 관계 / 제약조건

 - 개체-관계 모델/다이어그램



- 관계형 데이터 모델

 - 관계형 데이터 구조

- 개체-관계 모델을 관계 모델로 변환

릴레이션(Relation) R

□ 수학적 정의

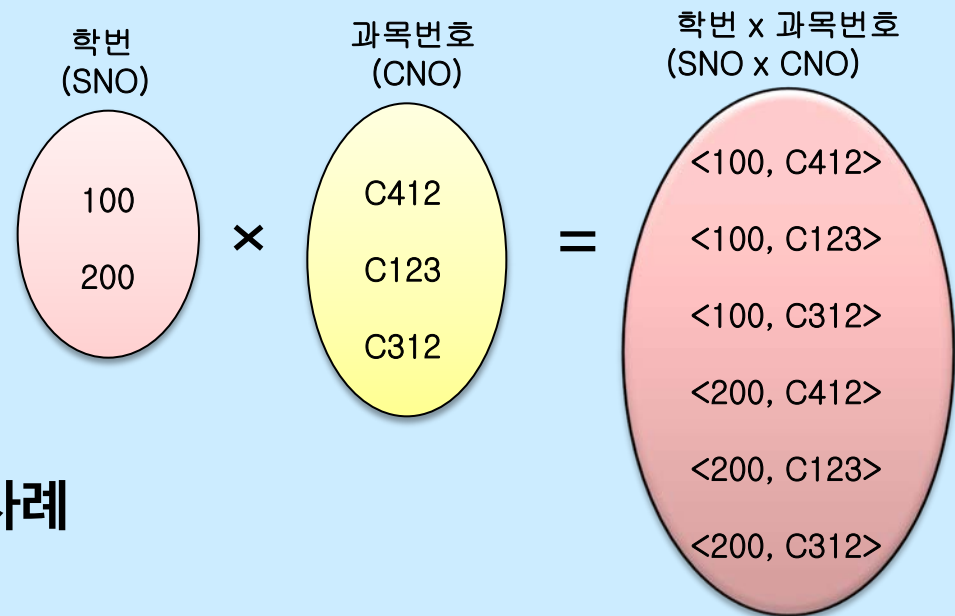
□ 릴레이션 R: 카티션 프러덕트의 부분집합

$R \subseteq D_1 \times D_2 \times \dots \times D_n$ 즉, n -튜플 $\langle v_1, v_2, \dots, v_n \rangle$ 의 집합

D_i : i 번째 도메인 $v_i \in D_i, i = 1, 2, \dots, n$

n : R의 차수(degree: 일차, 이차, 삼차, ..., n 차)

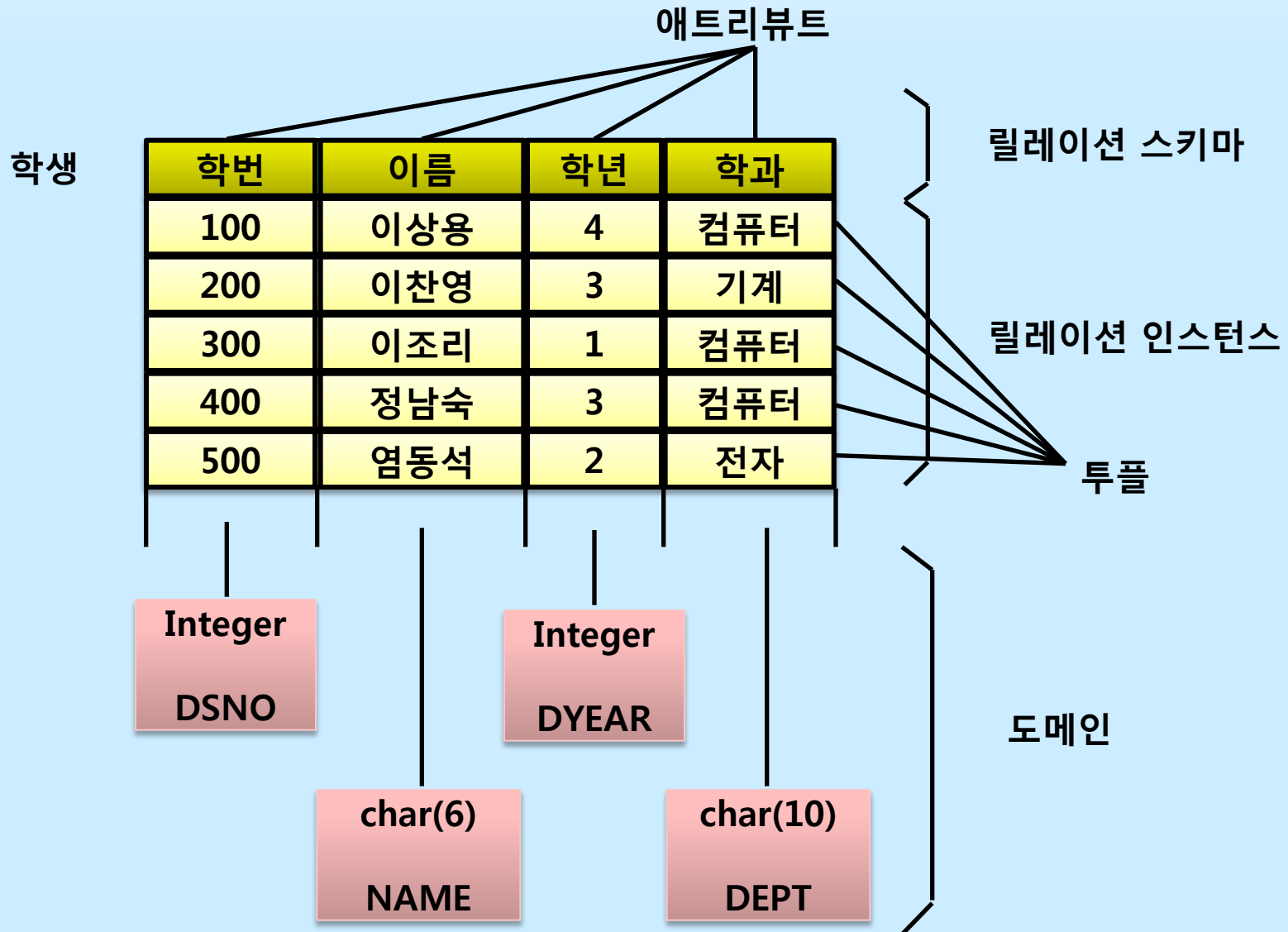
튜플의 수: 카디날리티(cardinality)



□ 개념적 정의

릴레이션 스키마 + 릴레이션 사례

릴레이션 스키마와 사례



릴레이션의 특성

□ 튜플의 유일성

- 릴레이션 = 서로 다른 튜플들의 "집합"

□ 튜플들의 무순서

- 릴레이션 : 추상적 개념
- 테이블 : 구체적 개념

□ 애트리뷰트들의 무순서

- 릴레이션 스키마 → 애트리뷰트들의 "집합"
- 튜플 : <attr.value>쌍의 집합

□ 애트리뷰트의 원자값(atomic value) - 분해 불가능

키(key) - I

- 릴레이션 $R(A_1, A_2, \dots, A_n)$ 의 속성집합 $K=\{A_i, A_j, \dots, A_k\}$
 - 유일성(uniqueness)
 - 각 튜플의 $K (= \{ A_i, A_j, \dots, A_k \})$ 의 값 $\langle V_i, V_j, \dots, V_k \rangle$ 은 유일
 - 최소성(minimality)
 - K 는 튜플을 유일하게 식별하기 위해 필요한 애트리뷰트로만 구성
- 수퍼키 (super key)
 - 유일성은 만족하지만 최소성은 만족하지는 않는 애트리뷰트의 집합
- 후보키(candidate key)
 - 유일성과 최소성을 모두 만족하는 애트리뷰트의 집합
- 기본키 (primary key)
 - 후보키(candidate key)중 데이터베이스 설계자가 지정한 하나의 키
 - 각 튜플에 대한 기본키 값은 항상 유효(no null value)해야 함
- 대체키 (alternate key)
 - 후보키 중에 기본키를 제외한 나머지 후보키

키(key) - II

□ 외래키(foreign key)

□ 릴레이션 R_1 에 속한 애트리뷰트 집합 FK가
릴레이션 R_2 의 기본키일 때 FK는 R_1 의 외래키

□ (FK의 도메인) = (R_2 의 기본키의 도메인)

□ FK의 값은 R_2 에 존재하는 값이거나 null이다.

□ $R_1 \neq R_2$ 인 경우

교수 (교수번호, 교수이름, 학과번호, 직급)

학과 (학과번호, 학과이름, 학과장교수번호, 학생수)
PK FK

학생 (학번, 이름, 학년, 학과)

과목 (과목번호, 과목이름, 학점, 학과, 담당교수)

등록 (학번, 과목번호, 성적)
FK FK

□ $R_1 = R_2$ 인 경우

교수1 (교수번호, 교수이름, 학과번호, 학장교수번호)
PK FK

무결성 제약(Integrity Constraints)

□ null 값

- 정보 부재를 명시적으로 표현하는 특수한 데이터 값
- 알려지지 않은 값(unknown value) 또는 해당 없음(inapplicable)

□ 개체 무결성(entity integrity)

- 기본키 값은 언제 어느 때고 null값을 가질 수 없음

□ 참조 무결성(referential integrity)

- 외래키의 값은 참조된 릴레이션의 기본키 값이거나 null임

□ 데이터베이스 상태 (database state)

- 어느 한 시점에 데이터베이스에 저장된 데이터 값 인스턴스
- 데이터베이스 상태 변화: 삽입, 삭제, 변경 연산
- DBMS는 데이터베이스 상태의 변화에도 항상 무결성 제약을 만족 시켜야 함

배울 내용

- 데이터 모델의 개념
- 개체-관계 데이터모델
 - 개체 / 관계 / 제약조건
 - 개체-관계 모델/다이어그램
- 관계형 데이터모델
 - 관계형 데이터 구조
- 개체-관계 모델을 관계 모델로 변환



관계형 스키마로의 변환(I)

□ E-R 스키마를 테이블로 변환

- 주 키는 엔티티 집합과 관계 집합이 데이터베이스의 내용을 표현하는 테이블로 균일하게 나타나도록 함
- E-R도에 상응하는 데이터베이스는 테이블의 모임으로 표현
- 각 엔티티 집합과 관계 집합에 대해 그에 상응하는 이름이 할당된 유일한 테이블이 존재
- 각 테이블에는 고유한 이름을 가진 여러 개의 열(일반적으로 애트리뷰트에 대응함)을 가진다.
- E-R도를 테이블 형식으로 변환하는 것이 E-R도로 부터 관계형 데이터베이스 설계를 유도하는 기본

관계형 스키마로의 변환(II)

□ 개체 집합을 테이블로 표현

□ 강 엔티티 집합은 같은 애트리뷰트를 가진 테이블로 변환

customer-name	social-security	c-street	c-city
Jones	321-12-3123	Main	Harrison
Smith	019-28-3746	North	Rye
Hayes	677-89-9011	Main	Harrison

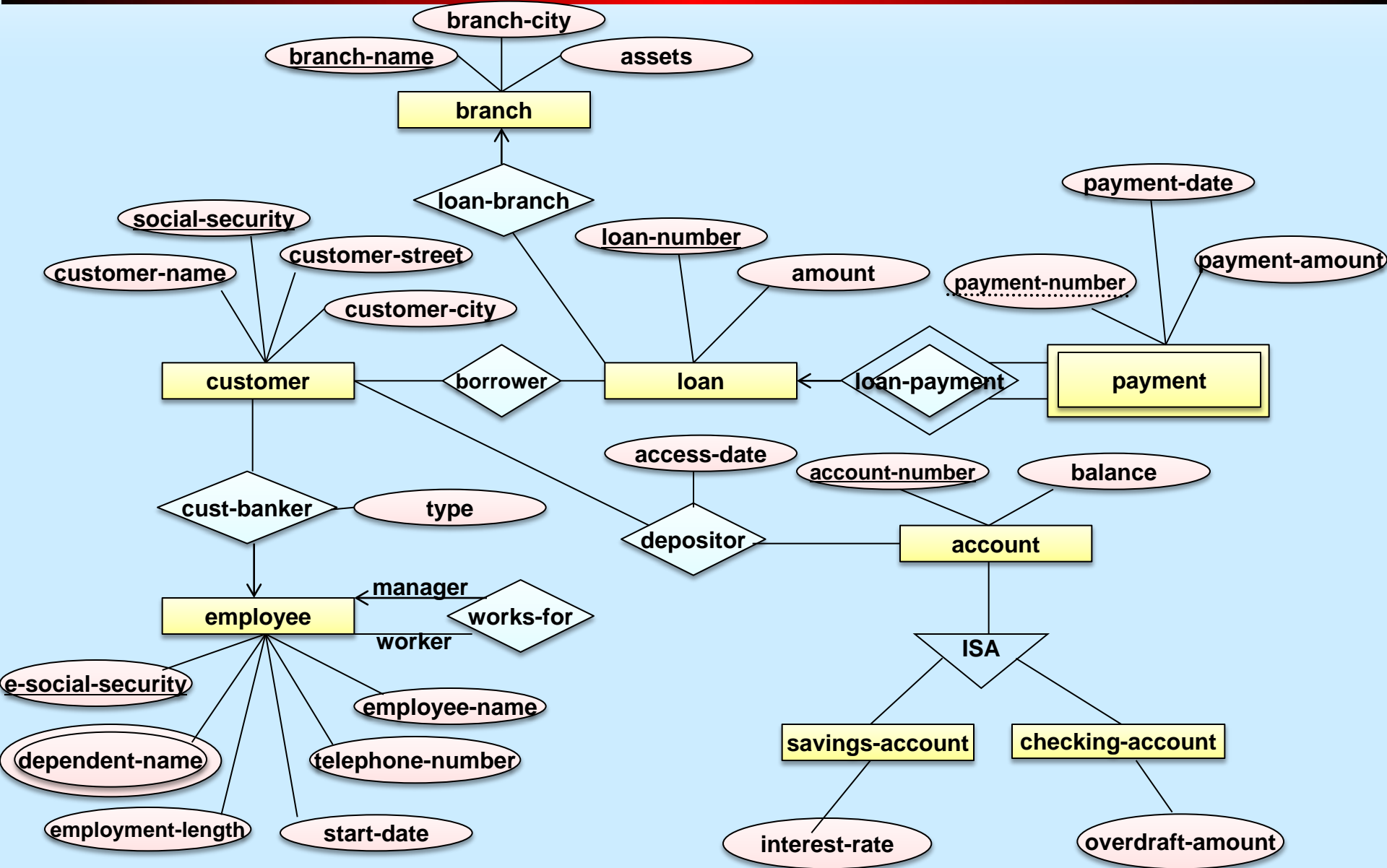
customer 테이블

□ 약 엔티티 집합은 강 엔티티 집합을 식별하는 주 키의 행을 포함한 테이블로 변환

loan-number	payment-number	payment-date	payment-amount
L-17	5	10 May 1996	50
L-23	11	17 May 1996	75
L-15	22	23 May 1996	300

Payment 테이블

E-R 다이어그램의 예 - 은행기관



관계형 스키마로의 변환(III)

□ 관계 집합을 테이블로 표현

- 다 대 다 관계 집합은 참여하는 두 엔티티 집합의 주 키와 관계 집합의 어떤 설명 애트리뷰트에 대한 열을 가진 테이블로 표현

social-security	account-number	access-date
...

depositor 테이블

- 약 엔티티 집합에 그를 식별하고 있는 강 엔티티 집합을 연결하는 관계 집합에 상응하는 테이블은 중복
 - payment 테이블에는 loan-payment 테이블에 나타난 정보 (즉, loan-number와 payment-number 열)를 이미 가지고 있음

관계형 스키마로의 변환(IV)

□ 일반화의 테이블 표현

□ 방법1 : 일반화 엔티티 account에 대한 테이블 작성

테이블	테이블 애트리뷰트
<i>account</i>	<i>account-number, balance, account-type</i>
<i>savings-account</i>	<i>account-number, interest-rate</i>
<i>checking-account</i>	<i>account-number, overdraft-amount</i>

□ 방법2 : 일반화되는 각 엔티티 집합에 대한 테이블 작성

테이블	테이블 애트리뷰트
<i>savings-account</i>	<i>account-number, balance, interest-rate</i>
<i>checking-account</i>	<i>account-number, balance, overdraft-amount</i>

E-R 집합으로부터 키의 결정

□ 강 엔티티 집합

- 엔티티 집합의 주 키는 릴레이션의 주 키

□ 약 엔티티 집합

- 릴레이션의 주 키는 강 엔티티 집합의 주 키와 약 엔티티 집합의 구별자의 합집합으로 구성

□ 관계 집합

- 관련된 엔티티 집합의 주 키의 합집합은 릴레이션의 수퍼 키

□ 이진 다 대 다 관계 집합에서는 수퍼 키가 또한 주 키

□ 이진 다 대 일 관계 집합에서는 “다”쪽 엔티티 집합의 주 키가 릴레이션의 주 키

□ 일 대 일 관계 집합에서는 어떤 쪽의 엔티티 집합의 주 키도 릴레이션의 주 키가 될 수 있음

- ❑ 데이터 모델의 개념
- ❑ 개체-관계 데이터모델
 - ❑ 개체 / 관계 / 제약조건
 - ❑ 개체-관계 모델/다이어그램
- ❑ 관계형 데이터모델
 - ❑ 관계형 데이터 구조
- ❑ 개체-관계 모델을 관계 모델로 변환



다음 배울 내용 : 관계형 데이터 모델 [관계 대수]