

ERD

E-R 모델 기본



DB는 개체(entity)들의 집합과 개체들 간의 관계로 모델링할 수 있다.

- **Entity/ Attribute / Entity Set**

- **Entity**

- : **Entity**란, 다른 객체와 구별되어 존재하는 하나의 객체를 뜻한다. 예로, 사람이나 책처럼 구체적인 것이 될 수도 있고, 수업이나 비행기 예약처럼 추상적인 것도 entity가 될 수 있다.

- **Attribute**

- : 이러한 entity는 **Attribute**를 가지고 있다. Attribute는 값을 가지고 있고, 각 entity는 attribute에 대한 자신만의 값을 가진다. attribute는 어떤 entity 집합의 구성원들이 공통적으로 가지는 특징을 표현해준다. 해당 집합의 attribute들의 일부로 각 entity를 고유하게 식별할 수 있게 된다.

- **Entity Set**

- : **Entity Set**은 각 attribute를 가지고 있는 entity들의 집합이다. (table과 비슷하다고 볼 수 있다.)

- **Relationship / Relationship Set**

- **Relationship**

- : **Relationship**이란, entity 간 의미있는 연관성을 의미한다. 즉, entity들이 서로 어떻게 연관되어 있는 지를 말한다.

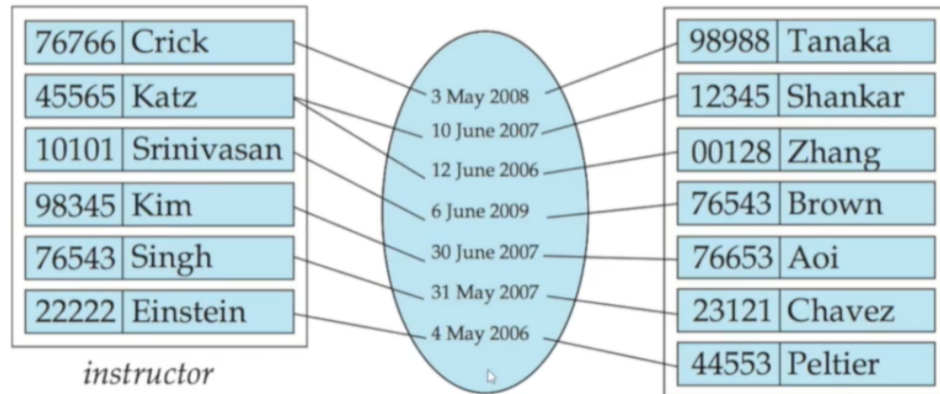
- **Relationship Set**

- : **Relationship**은 각 entity 간의 관계를 뜻하고, **Relationship Set**은 이런 Relationship의 집합을 뜻한다.

- **Relationship Set의 Attribute**

- : Relationship set 역시 Attribute를 가질 수 있다.

- An **attribute** can also be property of a relationship set.
- For instance, the *advisor* relationship set between entity sets *instructor* and *student* may have the attribute *date* which tracks when the student started being associated with the advisor



instructor(좌)와 student(우) entity들 간에는 advisor(중간)라는 Relationship set을 가지는데, advisor는 Date라는 Attribute를 가진다.

◦ Relationship Set의 Key

: Relationship set에도 각 Relation을 구분하기 위한 Key 개념이 존재한다. Relationship Set의 Key는 참여하는 entity Set의 Key들을 사용하여 만들세 된다. 위의 경우, student의 s_id와 instructor의 i_id의 조합이 advisor라는 Relationship set의 superkey로 선택한 예시이다.

◦ Relationship Set의 Degree 개념

: Relationship set에는 **Degree** 개념이 있다. 해당 relationship set에 참여하는 entity set의 개수를 말한다. 직전까지 살펴봤던 instructor와 student 간의 advisor 관계는 두 개의 entity set만 참여하므로 **binary relationship** 이라 한다. 대부분의 관계는 binary로 되어 있다.

• Attribute

: entity는 **Attribute** 의 집합으로 표현할 수 있다. attribute는 해당 entity set의 모든 멤버가 가지고 있는 특성을 말한다.

• Domain

: **Domain** 이란, 각 attribute에 허용되는 값들의 집합을 의미한다.

• Attribute type

- **Simple vs Composite**

:

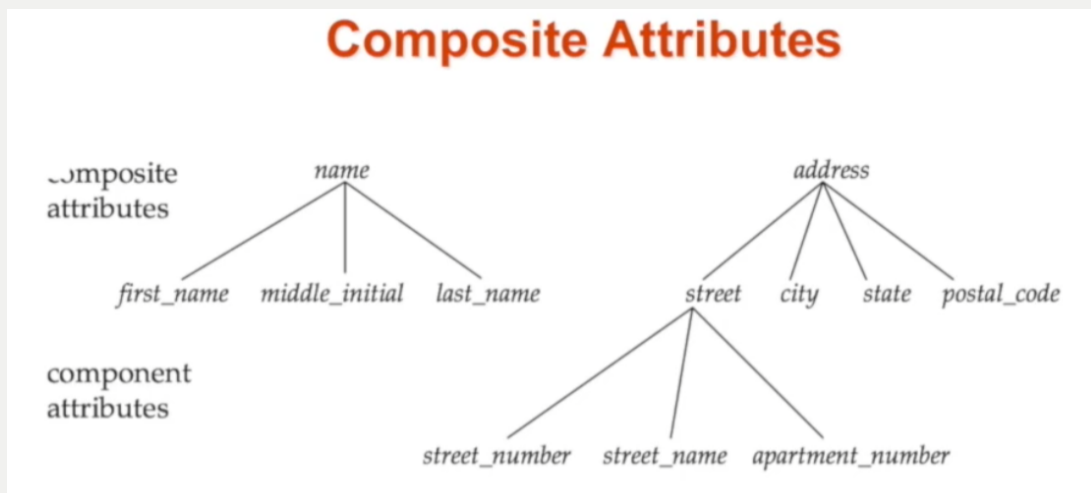
Simple attribute는 더 이상 작은 단위로 나뉘지지 않는 값을 가지는 경우를 말하고, 반대로 Composite attribute는 더 작은 단위 / 속성으로 나뉘질 수 있다.

- **Single-valued vs Multi-valued**

: single-valued attribute는 특정 entity에 대한 하나의 값만 가지고, Multi-valued는 한 번에 여러 개의 값을 가질 수 있는 경우이다. ex) 연락처 attribute에는 집전화/ 휴대폰 등 여러 개가 올 수 있다.

- **Derived**

: 직접 저장되지 않고 필요할 때마다 다른 attribute를 이용하여 계산되는 경우이다.



- **Composite Attribute 예시**

:

Composite Attribute는 경우에 따라 해당 속성 값의 전체 혹은 일부만을 참조하고 싶을 때 유용하다. 예로, name이라는 attribute는 풀네임으로 필요할 때도 있지만 first name만이 필요한 경우도 있다. 또한, address의 예처럼 계층 구조를 가질 수 있어 모델링을 더 명확하게 할 수 있고 연관된 attribute들을 하나의 그룹으로 묶을 수 있다는 장점이 있다.

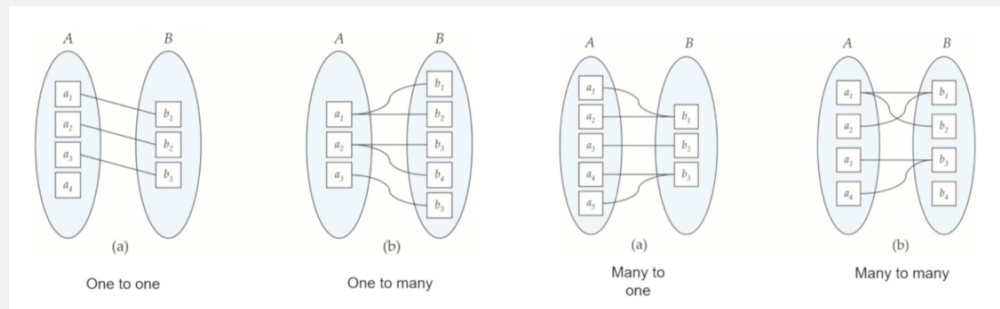
E-R 모델의 2가지 제약



1. Mapping Cardinality 제약

• Mapping Cardinality 란?

: Mapping Cardinality 는 개별 entity들이 매핑되는 개수를 말한다.



• 유형별 Relationship Set Key 결정

: 이전에 다룬 것처럼 Relationship Set에도 Key 개념이 존재한다. Relationship Set의 Key 는 참여하는 entity set의 Key들을 사용하여 만들게 되는데, 이때, Mapping Cardinality 에 따라 key 결정 방법이 다르다.

Mapping Cardinality는 위 그림처럼, 두 개의 entity Set만 참여하는 binary relationship을 기술하는데 가장 유용하게 사용된다.

binary relationship에는 4가지 유형이 있으며, 각 유형별 candidate key 를 결정하는 방법을 알아보자.

◦ Many to Many

: Many to Many Relationship의 경우 advisor의 primary key는 각 entity set의 primary key의 합집합이 될 수 있다.

◦ Many to One / One to Many

: Many 쪽에 있는 기본키를 Relationship의 기본키로 결정하게 된다. Many 쪽의 entity 간 구분이 되어야 하기 때문이다.

◦ One to One

: 두 entity set 중 어느 쪽의 기본키를 Relationship의 기본키로 사용하더라도 모든 relationship을 구분할 수 있다.

2. Participation 제약

: 참여에 관한 제약조건으로 전체 참여와 부분 참여가 있다.

- **Total participation (전체 참여)**

: 모든 entity가 적어도 하나의 Relationship에 참여하는 것

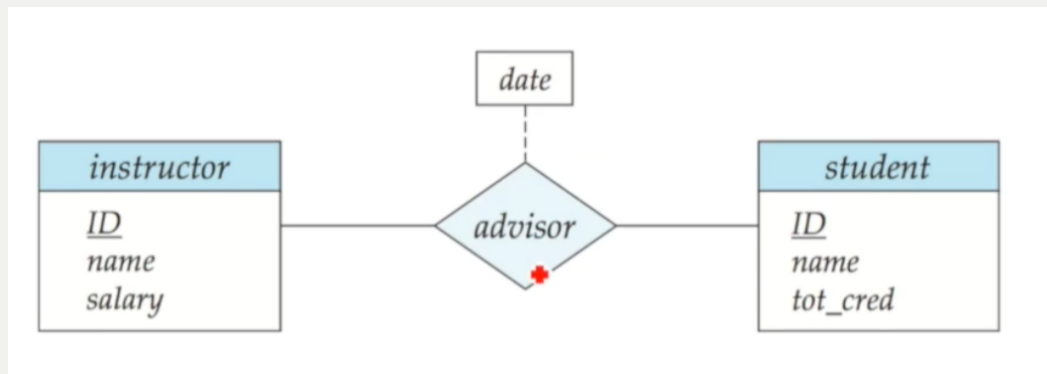
- **Partial participation (부분 참여)**

: 일부 entity는 참여하지 않을 수 있음

E-R 다이어그램



- **기본 표현**



: instructor와 student는 entity set이고, 박스 내부는 attribute들이다. 이 중 밑줄친 것은 primary key를 나타낸다. 종간의 마름모는 relationship을 나타내며, 점선 date는 relationship의 attribute를 나타낸다