

정성우_데이터 모델링

데이터 모델링의 이해

데이터 모델링이란?

```
Input
( , , erd... )
```

WHY ?

- 시스템의 구조(Data)와 업무 프로세스(Process)를 명세화 할 수 있다
- 현재 또는 원하는 시스템을 가시화하여 시스템을 구축하는 기본 틀을 제공한다
- 기업이 비즈니스의 목적을 바꾸지 않는 한 비즈니스에서 관리하는 데이터는 거의 연속적으로 변화되지 않는다
- 가능성 실험 - 문제의 다양한 해결 방안을 실험
- 생산성 - 경제적인 효율과 효율성
- 이식성 - 비슷한 모델에 대해 바로 적용

데이터 모델링 단계

```
:
:
( ex )
: , (DBMS )
:
```

논리적 데이터 모델링이란

- 데이터베이스 개발과정의 첫 단계
- 전략 수립 및 분석 단계에서 실시
- 요구되는 (분석한) 정보를 실체(엔티티)와 관계로 모델링하는 것

데이터 모델링의 3요소

1. Entity : 업무가 다루는 사물(실체)
2. Relation : 사물들 사이에 존재하는 연관(관계)
3. Attribute : 사물이 지니는 상세한 특성(속성)

개념(Concept)	타입(Type)	어커런스(Occurrence)
사물	실체(Entity)	인스턴스(Instance)
연관성	관계(Relation)	연관 (assoication)
특성	속성(Attribute)	값 (value)

Entity 란

데이터베이스에서 표현되어야 하는 구별 가능한 객체

각각의 인스턴스들의 집합을 나타낸다

Entity의 특성

- 구현 조직에서 저장, 관리하고자 하는 의미 있는 정보를 관리

- Entity명은 현실 세계의 객체 및 정보를 대표
- 실체가 포함하는 정보에 대해 무엇, 왜, 범위에 대해 명확히 표현할 수 있어야 함

정확하게 바인더리가 정해져 있어야 한다

예제 : Entity 명확화

- 개인정보를 관리 : 개인만 회원으로 포함할 것인가 법인도 포함할 것인가?
- 직원정보를 관리 : 정규직만 포함할 것인가 비정규직도 포함할 것인가?
- 고객을 관리 : 우리 상품을 구매(혹은 가입, 계약)하지 않은 사람도 고객인가?
- 연예인을 관리 : 가수, 연기자, 개그맨으로 나눌 때 낸시랭은 연예인인가 아닌가?
- 우리 회사 직원들도 상품을 구매할 때 고객으로 볼 것인가?

Relation이란?

- 엔티티 상호간에 어떻게 연관되어 있는지를 파악하여 연관관계를 표현
 1. 관계의 식별성 (Identification)
 2. 선택성 (Optionality)
 3. 관계형태 (Degree)

ex) 모든 Entity1은 단 하나/ 단하나 이상 의 Entity2와/에 반드시 관계 해야 한다/ 관계 일수도 있다

- 단 하나/ 단하나 이상 : 관계형태 (Degree)
- 반드시 관계 해야 한다/ 관계 일수도 있다 : 선택성 (Optionality)

식별성 (Identification)

1. (Identification Relation)
 - | ()
 - (UID) (UID) .
2. (Non-Identification Relation)
 - |
 - (UID) (UID)

선택성 (Optionality)

1. 필수적 (Mandatory, Must)
 - — (실선) 표시
 - 자식 엔티티에 데이터를 입력하기 전에 부모 엔티티에 적어도 한건 이상의 데이터가 있어야만 하는 경우
2. 선택성 (Optional, May)
 - —— (점선) 표시
 - 자식 엔티티에 데이터를 입력하기 전에 부모 엔티티에 데이터가 존재하지 않아도 상관없는 경우.

기수성 (Degree)

1. 일
 - — 로 표시 (One and Only One)
2. 여러명
 - ∈ : 삼지창으로 표시 (One or Many)
- 표현하는방법
 - 제 1자로서 보는 관계
 - 관계명은 가능한 제 1자의 입장에서 규정
 - 다른 관계와 차별성 있게 구체적으로 지정

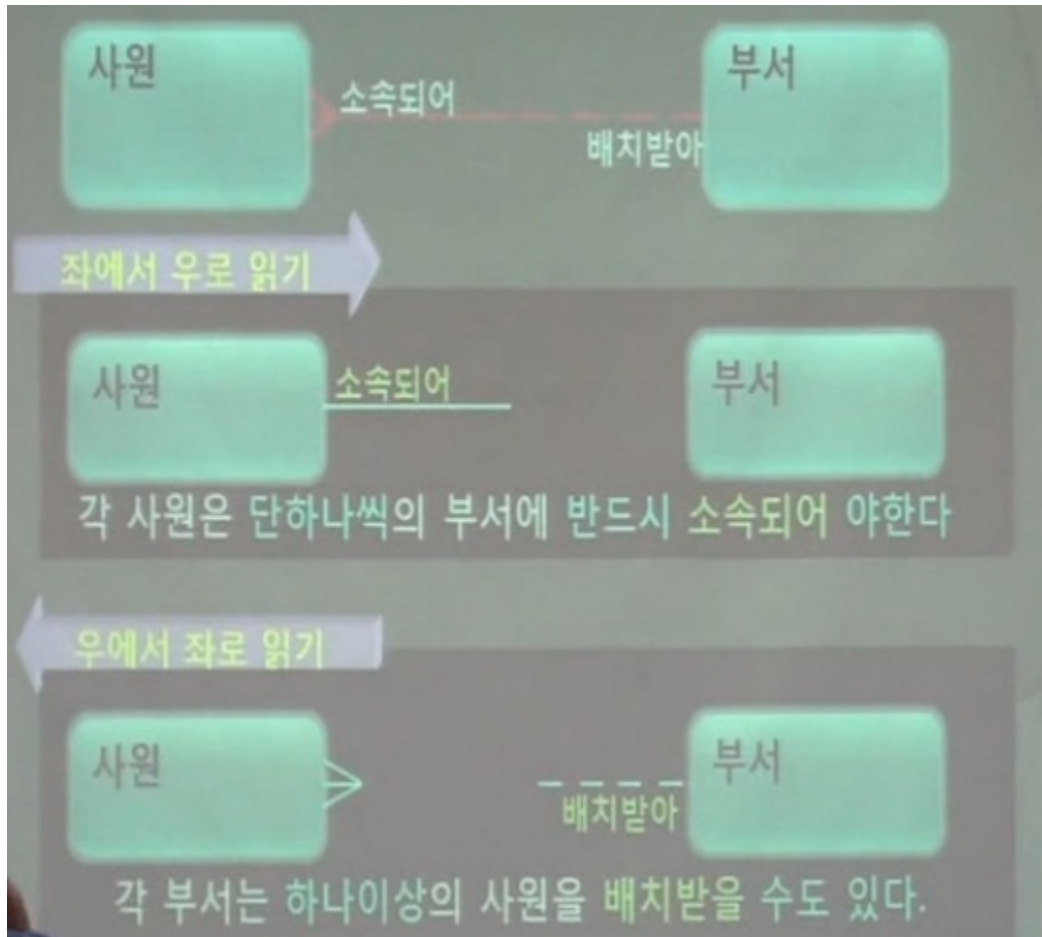
ex) 신랑과 신부를 나누어볼 때 신랑이보는 입장 신부가 보는 입장을 따로 구분
 - 제 3자로서 보는 관계
 - 제 3자의 입장에서 관계의 내용을 보다 명확히 파악
 - 관계를 규정할 때에는 반드시 제 1자의 입장에서

ex) 이 둘사이의 부부로서의 관계다
 - 역할 위주로 정의하는 경우
 - 주는 쪽(One)의 특정 집합이 서브타입으로 지정되어 있지 않은 경우 주체들을 구체적으로 명시할 때 사용

ex) 부서가 공사에 참여할 때 수행부서로써 참여할 수도 있지만 순익부서로써도 참여하는 관계
 - 내용 위주로 정의하는 경우
 - 관계를 맺는 집합은 동일하나 관계를 맺는 내용이 다를 때 사용

ex) 부서와 직원 사이에 부서는 현소속으로써 관리도 하지만 근무이력으로써 관리도 한다

- 릴레이션(Relation) 그리는 순서
 1. 두 엔티티(Entity) 사이에 선을 그린다
 2. 관계 명칭을 기록한다
 3. 선택사양(Optionality)를 표시한다
 4. 관계형태를 표시한다
- 릴레이션(Relation) 읽기

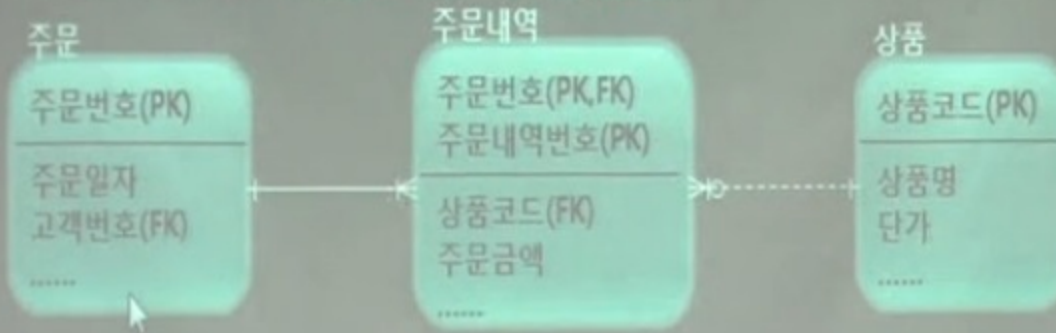


속성(Attribute) 이란?

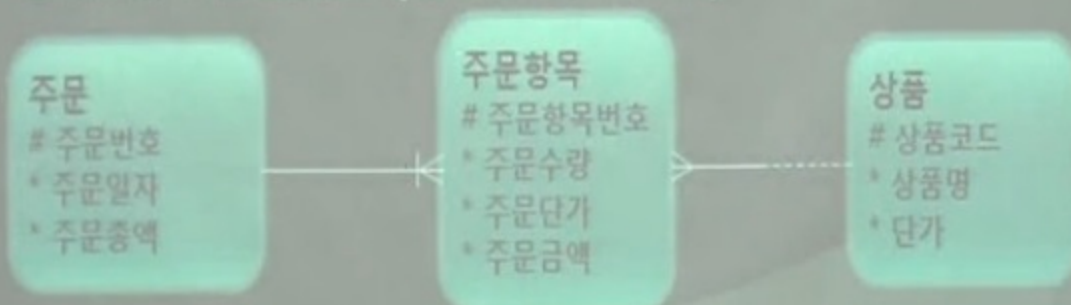
- 엔티티(실체, 사물, 개체)의 본질을 이루는 고유한 성질이나 특성으로서 우리가 관리하고자 하는 정보 항목들이다.
- 속성(Attribute)의 지정원칙
 1. 원자 단위인가?
 2. SINGLE VALUE 인가?
 3. 추출값(Derived Value) 인가?
 4. 보다 상세한 관리가 필요한가?
 - ex) 성명을 성과 이름으로, 전화번호를 지역번호, 국번, 개별번호로 나누는 것
- 추출 속성(Derived Attribute)
 - 개수, 합계, 최대, 최소, 평균, 계산값 등
 - 추출 속성의 기본이 되는 속성값에 따라 변경된다
- 속성(Attribute) 과 엔티티(Entity)의 비교
 - 속성이 자신의 속성을 가지면 엔티티(실체)
 - 실체는 속성을 갖지만 속성은 자신의 속성을 가질 수 없다

속성안에 다시 속성을 가질수있으면 엔티티로 빼는 것이 좋다
속성을 갖지만 자신안에 다시 속성을 가질 수 없다
- 속성(Attribute) 의 표현

IE Notation (정보 공학 방법론)



Barker Notation (CASE*Method)



- Notation 표기법 비교

▪ Notation 표기법 비교

IE Notation (정보공학방법론)	Many to One (M:1)			
	Many to Many (M:N)			
	One to One (1:1)			
Barker Notation (CASE* Method)	Many to One (M:1)			
	Many to Many (M:N)			
	One to One (1:1)			

식별자(Identifier) 이란?

- 현실에서의 식별 문제
 - 매우 유사한 특성을 가진 두 개의 사물을 어떻게 구분할 것인가
- 데이터베이스 내에서의 식별 문제
 - 테이블내의 여러 개의 행(row)들을 어떻게 구별할 것인가
- 표현의 문제
 - 테이블에서 실 세계의 사물이 표현된 것을 어떻게 알 수 있는가
- 식별자(Identifier)의 검증
 - 식별자를 구성하는 속성과 관계가 NOT NULL인가
 - 식별자를 구성하는 속성과 관계가 UNIQUE인가
 - 식별자를 구성하는 속성과 관계가 MINIMAL SET인가
- 인위적인 식별자
 - 식별자로 사용할 속성이 없거나 부적절한 경우 종종 인위적인 식별자가 사용됨
 - 식별자의 길이가 너무 긴데 자주 사용해야 하는 경우
 - 이미 실무에서 사용 중이거나 범용적인 속성을 사용할 것

정규화(Normalization) 이란?

‘ ‘ ‘ ‘

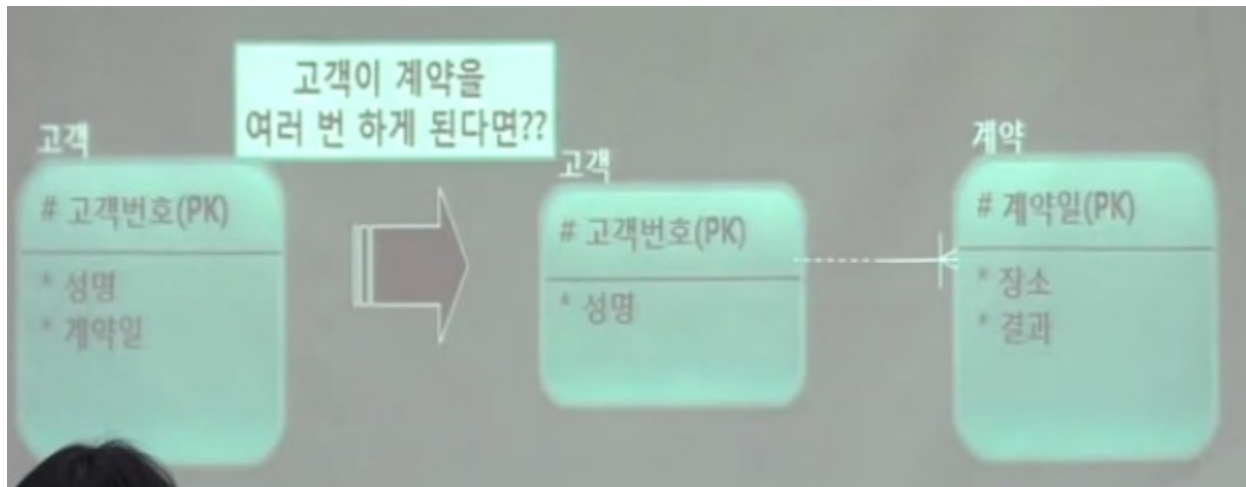
-
-
-
-
-

- 제1 정규형
 - 모든 속성은 반드시 하나의 값을 가져야 한다

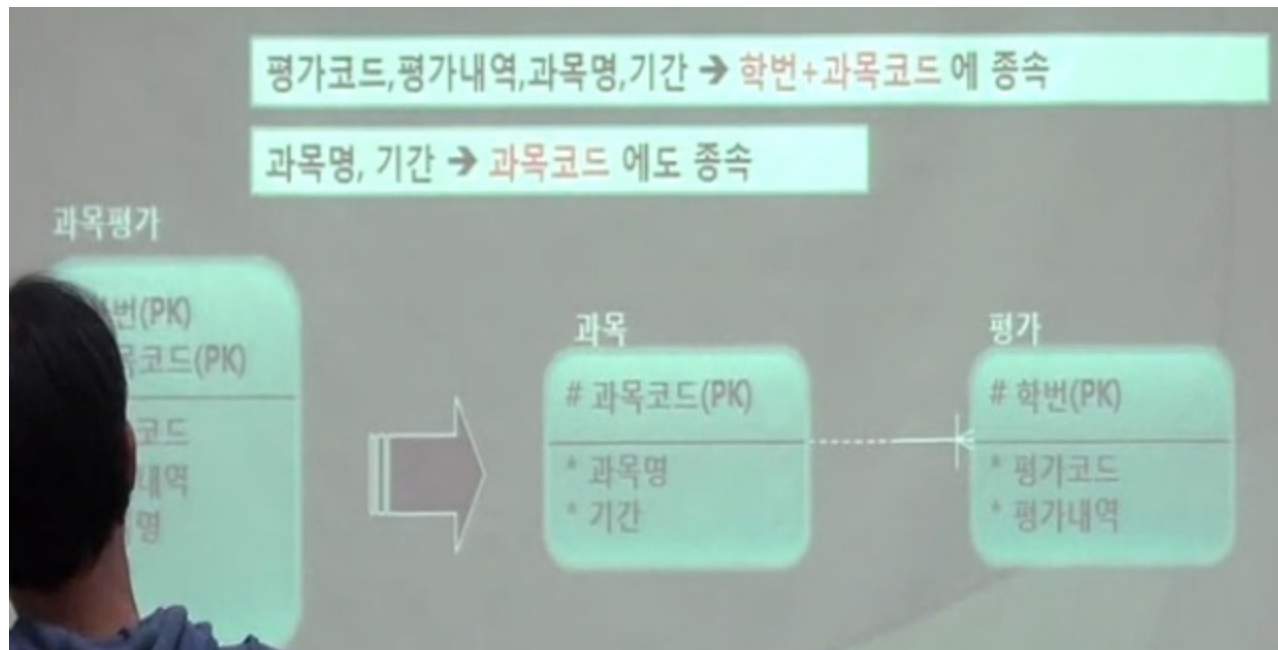
- 반복 또는 복수 값을 갖는 속성을 제거
- 제2 정규형
 - 모든 속성은 반드시 UID 전부에 종속되어야 한다
 - UID의 일부에만 종속되는 속성을 제거
- 제3 정규형
 - UID가 아닌 속성간에는 서로 종속될 수 없다
 - UID가 아닌 일반 속성에 종속적인 속성을 제거
 - 데이터 중복을 제거하기 위한 데이터베이스 설계의 일반적인 목표

예제

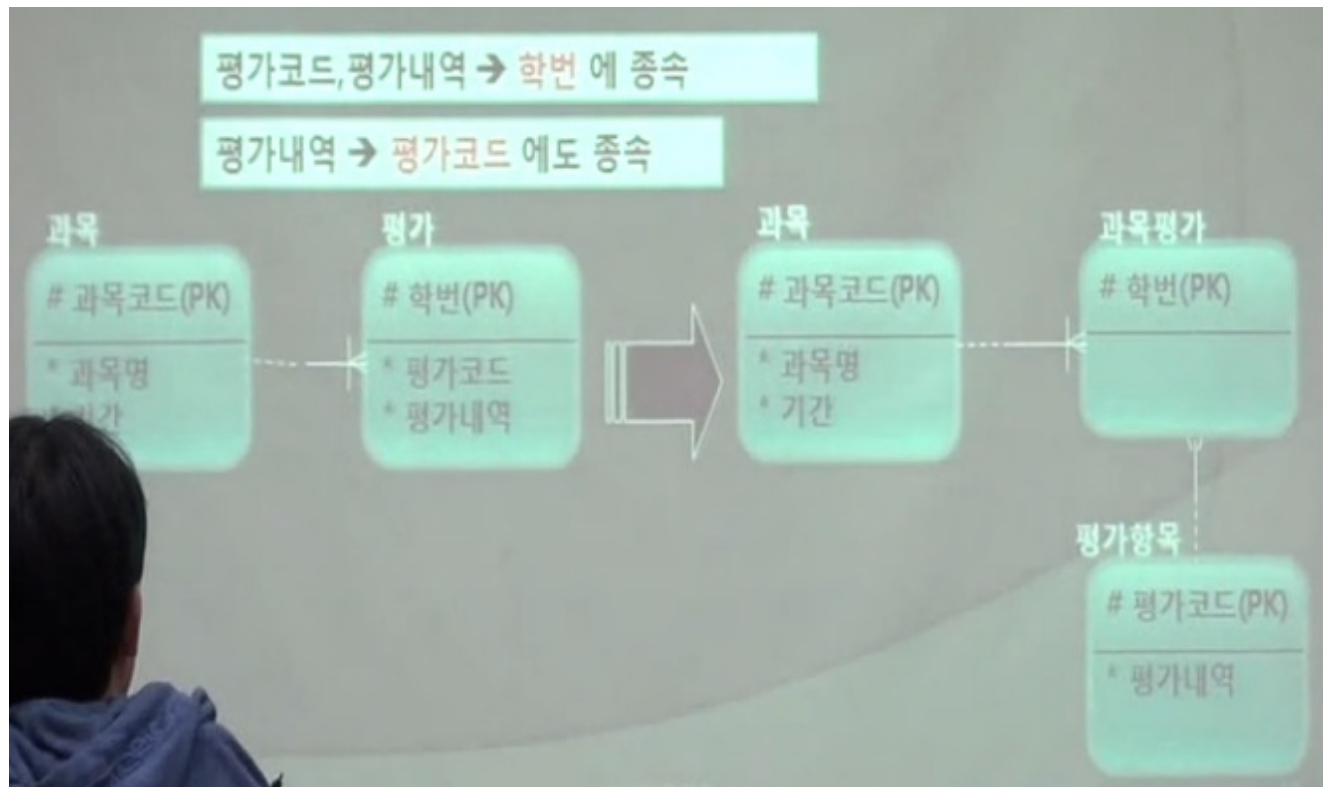
* 1
- 1 : N
ex) (PK) ,



* 2
- UID
ex) ..



* 3
-
ex)



* M : N
- / M:N
-
- M : N 1:N

반정규화(DeNormalization) 이란?

, , ,

* ?
* ?

- 반정규화(역정규화) 대상 조사
 - 범위 처리 빈도수 조사
 - 대량의 범위 처리 조사
 - 통계성 프로세스 조사
 - 테이블 조인개수
- 다른 방법 유도 검토
 - 뷰 테이블
 - 클러스터링 또는 인덱스 적용
 - 애플리케이션
- 정규화(역정규화) 적용
 - 테이블 반정규화
 - 테이블 병합
 - 1 : 1 관계의 테이블 병합
 - 1 : N 관계의 테이블 병합
 - 슈퍼타입 서브타입 테이블 병합
 - 테이블 분할
 - 수직 분할
 - 수평 분할
 - 테이블 추가
 - 중복 테이블 추가
 - 통계 테이블 추가
 - 이력 테이블 추가
 - 부분 테이블 추가
 - 속성의 반정규화
 - 컬럼 중복
 - 한 테이블에 대해 여러 개의 엔티티가 있을 때 조인 관계가 많을 때..
 - 파생컬럼의 생성
 - 관계의 반정규화

계층구조(Parent - Child) 이란?

ex) , , ...

- 순환 관계를 가진 구조
 - 구조에 알려진 레벨 제한이 없을 경우
 - 구조에 레벨 한계가 있지만 그 한계가 높을 경우
 - 구조의 인스턴스가 쉽게 위치를 변경할 수 있고 그로 인해 레벨이 변경되는 경우
 - 사용자가 제약 조건 유지를 선호할 경우

id	Parent	Name
id1	0	category1
id12	id1	category12
id13	id1	category13
id123	id12	category123
id124	id12	category124

- 비순환 관계를 가진 구조
 - 구조의 레벨 한계가 낮을 경우
 - 상위 레벨 변경 빈도가 낮을 경우
 - 데이터가 많고, 성능개선이 우선될 경우

Id	Level1	Level2	Level3
id1	id1		
id12	id1	id12	

id13	id1	id13	
id123	id1	id2	id3
id124	id1	id2	id4

네트워크 구조(Network, BOM) 이란?

), , , ..

- 특징
 - m : m 관계는 쌍에 대한 특정 정보를 보유
 - 교차 엔티티의 두 관계는 고유 식별자를 구성
 - 두 개의 관계는 상이한 Sub-type 의 엔티티를 참조