



데이터베이스

ER 다이어그램 표기법





학습목표

- ➔ 바커 표기법을 이용하여 개체-관계 모델을 작성할 수 있다.
- ➔ IE 표기법을 이용하여 개체-관계 모델을 작성할 수 있다.



학습내용

- ➔ 바커 표기법
- ➔ IE 표기법



바커 표기법



바커 표기법

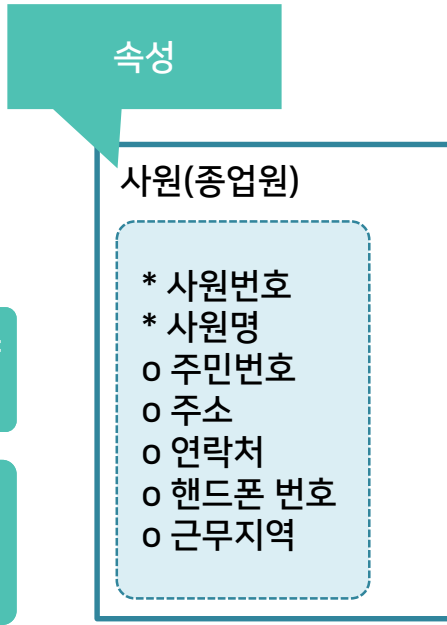
01 개체

- 기업에서 지속적으로 저장하고 관리해야 할 대상
- 하나의 관리 대상이 개체가 되기 위해서는 반드시 두 개 이상의 속성을 가져야 함
- 속성이 없는 실체는 존재할 수 없음
- 개체는 실제 세상에 있는 객체(Object)
 - 네 부분의 모서리가 둥근 형태인 소프트-박스(Soft-Box)로 표현
 - 하나 이상의 속성으로 구성



02 속성

- 하나의 개체에 종속되는 명사적 단어들을 의미
- 일반적으로 명사적 단어들 중에 구성 요소를 포함하고 있는 명사들은 개체가 되고 그렇지 못한 명사들은 속성이 됨
- 속성의 상태에는 2가지 종류가 있음



- *(Mandatory) ... 어떤 값을 반드시 저장해야 하는 경우
- o(Optional) ... 해당 속성에 어떤 값이 존재할 수도 있고 존재하지 않을 수도 있는 경우



바커 표기법



바커 표기법

03

식별자

- | 하나의 개체에 구성되어 있는 여러 개의 속성 중에 **개체를 대표할 수 있는 속성**
- | **하나의 개체는 반드시 하나의 식별자가 존재**
 - 식별자 = 키(Key)
 - 식별자: 논리 데이터 모델링 단계에서 사용
 - 키: 물리 데이터 모델링 단계에서 사용

| 식별자의 종류

본질 식별자

후보 식별자

대체(보조) 식별자

인조 식별자

실질 식별자



바커 표기법



바커 표기법

03

식별자

식별자의 종류

본질 식별자

- 속성들 중에서 집합의 본질을 명확하게 설명할 수 있는 **의미상의 주어**
- 의미상의 주어의 종류

인조 식별자 (Artificial Unique Identifier)	<ul style="list-style-type: none">• 사원번호, 상품번호처럼 집합을 식별하기 위한 임의의 유일 값을 사용
자신의 고유 속성과 부모로부터 물려받은 속성(릴레이션십)들	<ul style="list-style-type: none">• 내가 태어나기 위해서 절대적으로 존재했어야만 하는 본질 속성들에 해당하는 것

후보 식별자

- 각 **인스턴스를 유일하게 식별**할 수 있는 속성 또는 속성들의 조합
- 후보 식별자로 속성 집합을 선택하는 경우 개념적으로 유일해야 함



바커 표기법



바커 표기법

03

식별자

식별자의 종류

대체(보조) 식별자

- 원래의 식별자를 대신할 수 있는 또 다른 속성들이나 릴레이션십을 지칭

예 | 사원 개체에 공식적으로 부여된 식별자(실질 식별자)의 경우



사원 개체에 공식적으로 부여된 식별자(실질 식별자)는 사원번호임
→ 만약, 주민등록번호 속성이 유일한 값을 가지면서 필수적(Mandatory)으로 정의되었다면, 비록 공식적인 식별자는 아니지만 식별자로서의 역할을 할 자격은 충분히 갖추고 있음

- 보조 식별자는 여러 참조 개체 중에서 원래의 식별자보다 보조 식별자로 연결을 맺는 것이 자신에게는 훨씬 유리한 경우에 의미가 있음



바커 표기법



바커 표기법



식별자

식별자의 종류

인조 식별자

- 식별자 확정시 기존의 본질 식별자를 그대로 **실질 식별자로 인정할 수 없는 여러 가지 상황**이 발생했을 때, 전부 혹은 일부를 임의의 값을 가진 속성들로 대체하여 새롭게 구성한 식별자

» 예 | 사원 개체에 이미 존재하고 있는 속성 중에서 원래의 본질 식별자를 찾아야 할 경우



원래의 본질 식별자를 찾아야 할 경우
주민등록번호가 될 것임
→ 주민등록번호라는 속성은 너무 길고 관리상 여러 가지 문제점이 발생
→ 새롭게 사원번호라는 임의의 값을 가진 인조 속성을 영입하여 공식적인 식별자 자리까지 부여 받음

실질 식별자

- 인스턴스를 식별하기 위해 **공식적으로 부여된 식별자**
- 본질 식별자나 인조 식별자 모두가 실질 식별자가 될 수 있음



바커 표기법

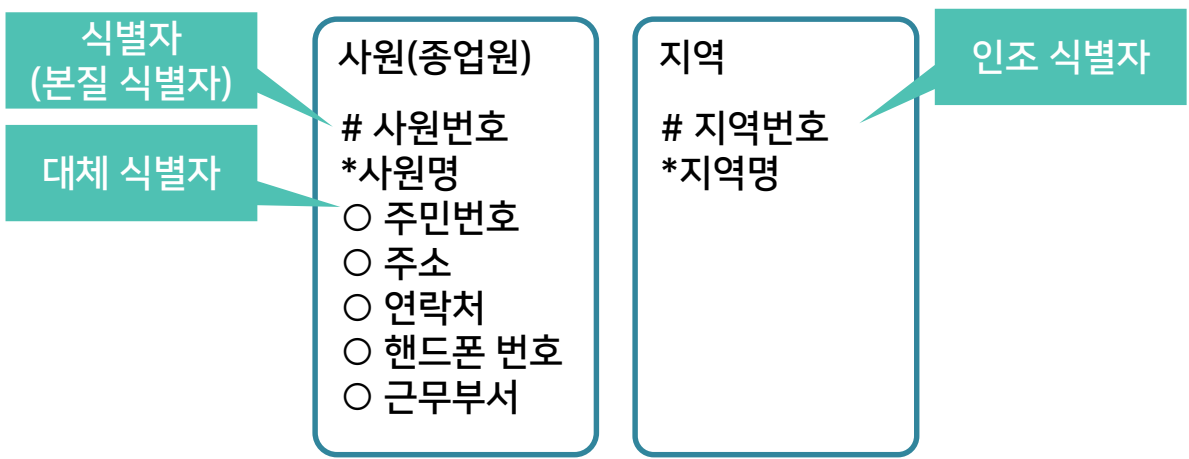


바커 표기법

04

식별자의 작성 방법

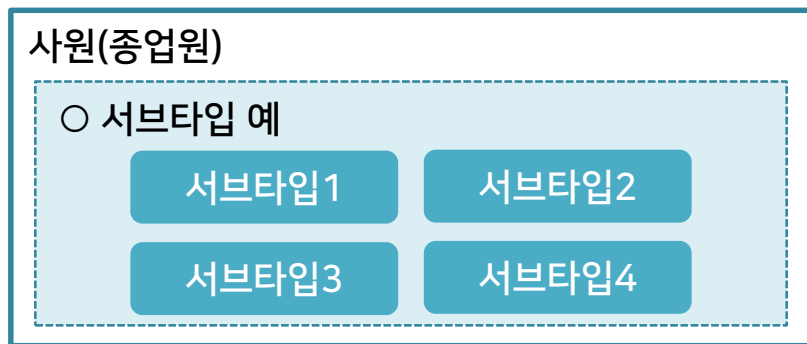
- 식별자 앞에는 # 기호를 표시하고 여러 개 # 기호를 반복적으로 표시



05

서브타입

- 바커 표기법에서는 슈퍼타입(Super-type) 안에 서브타입(Sub-type)을 상자로 나타냄
 - 장점: 다이어그램에서 공간을 적게 사용함
- 서브타입은 서브타입 간의 중복을 허락하지 않는 상호 배타적 관계





바커 표기법

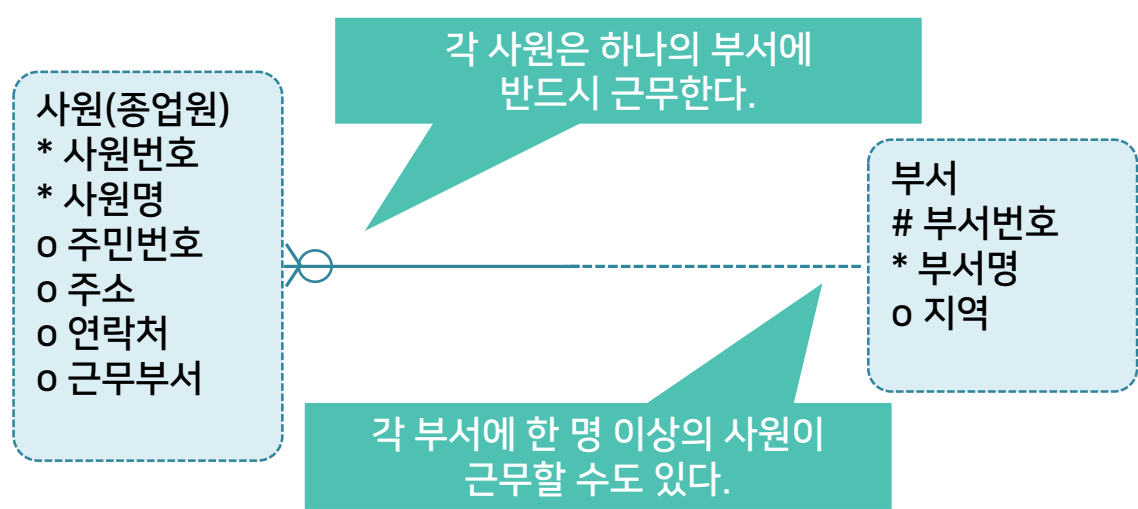


바커 표기법

06

관계

- 두 개의 개체 간에 조건을 표기한 후 해당 개체의 가까운 위치에 관계 명칭을 표기
- 관계는 실세계의 해당 개체에서 발생하는 동사적 단어들을 표기





바커 표기법



바커 표기법

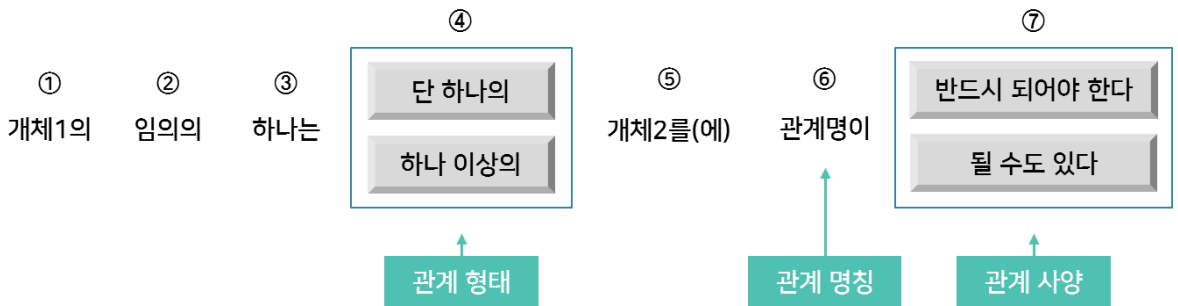
07

관계의 정의 방법

Ⅰ 관계를 규명할 때 가장 먼저 해야 할 일

- 한쪽 개체를 기준으로 **상대 개체와의 관계를 규명하고 다시 반대 방향으로 관계를 규명**

≫ 예 | 관계 구문의 예



개체1



관계를 규명할 두 개체 중에서 어느 한쪽이 주어가 되어 상대방과의 관계를 규명하고자 할 때 **주어가 되는 개체**

임의의



- 영어적 표현 **ANY**와 같다고 이해
- 집합 내의 개체 중에서 아무것이나 **하나(임의의 개체)**를 **선택하여 만족하는지를 증명하는 것**

하나는



- 주어 집합에 있는 '**임의의**' 또는 '**하나**'의 개체를 말함
- 이 하나가 어떤 형태로 상대방과의 관계를 가지는지를 검증

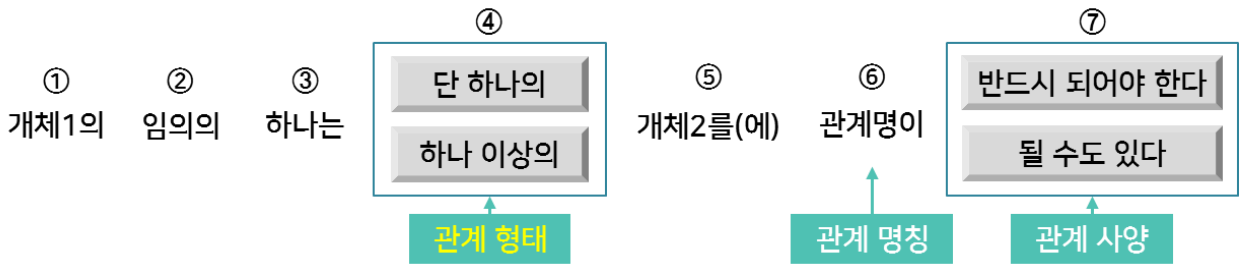


바커 표기법



바커 표기법

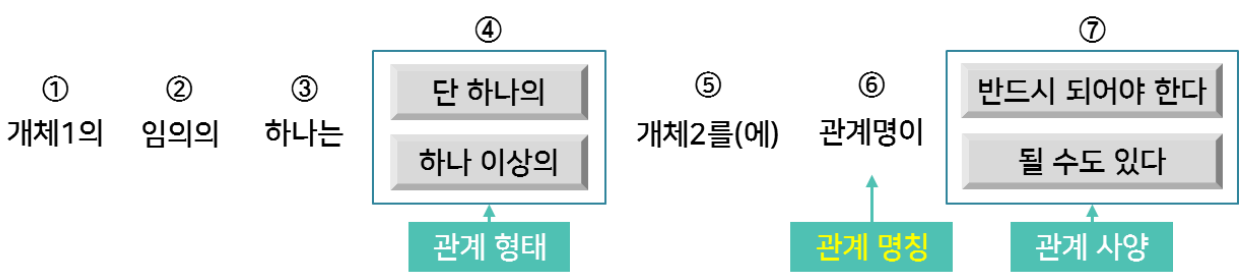
07 관계의 정의 방법



관계 형태



- 하나가 **몇 개의 상대방을** 가질 수 있는지를 **집중적으로 밝히는 것**
- 단 하나인 것을 증명하기 위해서 하나만 가지는 경우를 수없이 내세우는 것보다 **하나 이상인 경우를 한 가지만 제시하는 것**이 보편적인 방법



개체2



주어가 되는 개체가 바라 본 **상대방 개체**

관계명



관계 규명을 통해서 정의된 **관계 명칭**

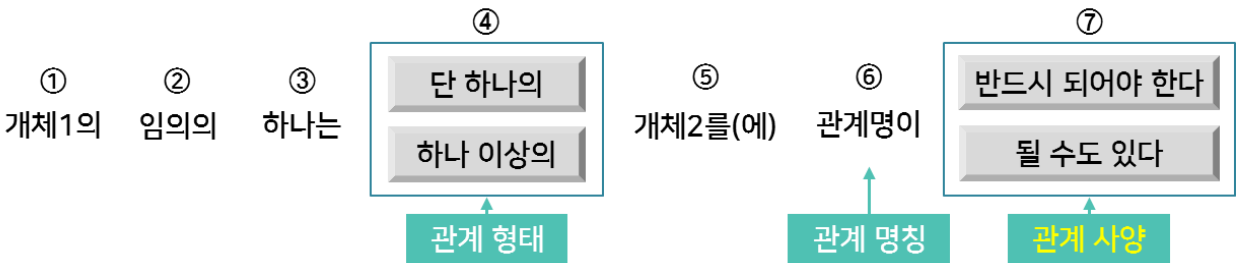


바커 표기법



바커 표기법

07 관계의 정의 방법



관계 형태 ...>

- 필수(실선), 선택(점선)의 모습을 결정하는 과정
- '반드시 관계명이어야 한다'와 '관계명일 수도 있다'를 결정하기 위해서 관계명을 만족하는 경우를 아무리 많이 제시하더라도 없을 수도 있다는 한 가지 경우만 예를 들면 결정할 수 있음

08 개체 간의 관계

1:1 관계	<ul style="list-style-type: none"> A 개체에 존재하는 데이터 1개와 관계되는 B 개체에 존재하는 데이터의 개수도 1개인 개체 간의 관계
1:M 관계	<ul style="list-style-type: none"> A 개체에 존재하는 데이터 1개와 관계되는 B 개체에 존재하는 데이터의 개수가 여러 개인 개체 간의 관계
M:M 관계	<ul style="list-style-type: none"> A 개체에 존재하는 데이터 1개와 관계되는 B 개체에 존재하는 데이터의 개수가 여러 개이며, B 개체에 존재한 데이터 1개와 관계되는 A 개체에 존재하는 데이터의 개수도 여러 개인 개체 간의 관계



바커 표기법



바커 표기법

09

개체 간의 상관 관계

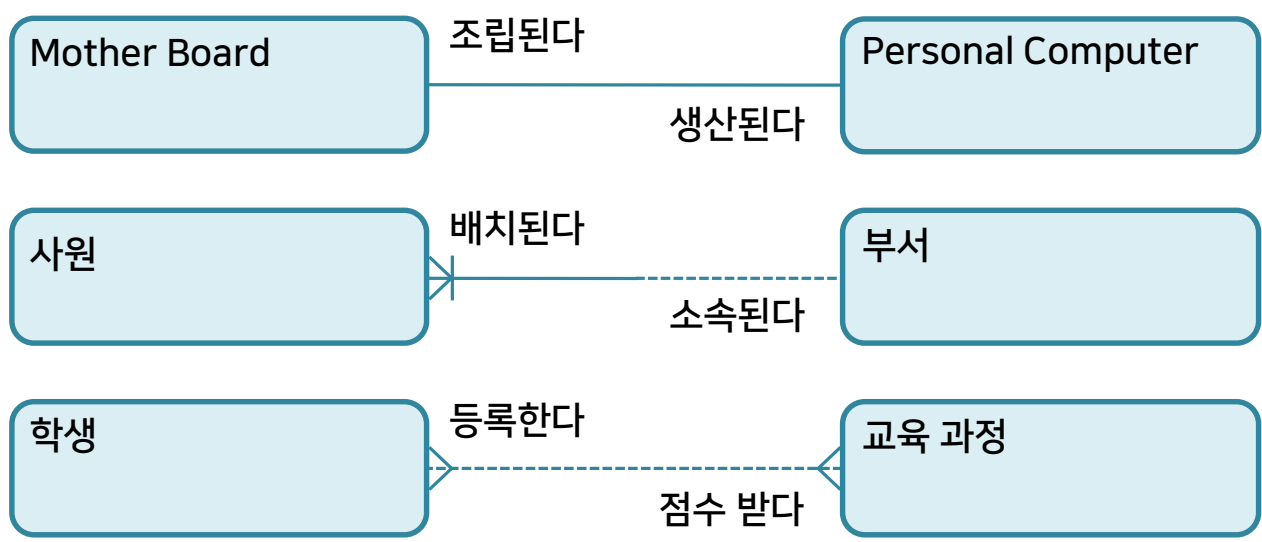
필수 조건

필수 사항은 **실선으로 표시**하고
상대 개체에 대해 해당 개체에
조건을 만족하는 개체가 반드시
존재할 경우에 표시

VS

선택 조건

선택 사항은 **점선으로 표시**하고
상대 개체에 대해 해당 개체에
조건을 만족하는 개체가 **존재할**
수도, **존재하지 않을** 수도 있을
경우 표시





바커 표기법

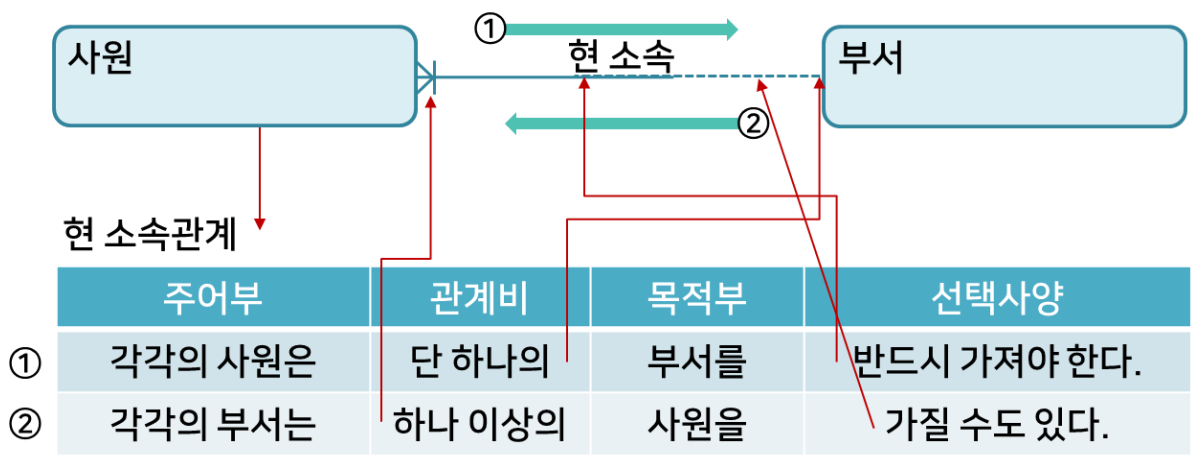


바커 표기법

10

관계의 표현

- 사원 개체가 주어일 경우 사원 개체 쪽에 가까이 있는 실선이 바로 **사원이 주어일 때 부서를 바라보는 관계를 설명한 것**





바커 표기법



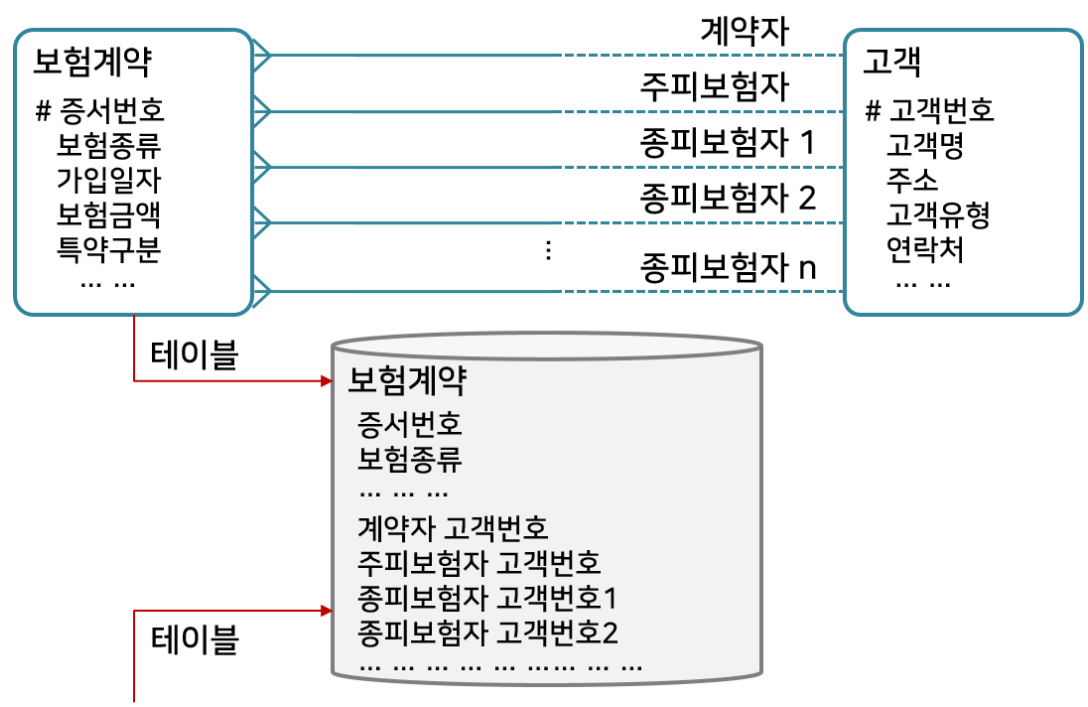
바커 표기법

10

관계의 표현

병렬식 관계

- 두 개체 사이에 존재하는 관계들을 별도의 관계로 간주함으로써 여러 개의 관계 선분이 나란히 병렬로 관계가 맺어지게 되는 경우





바커 표기법



바커 표기법

10

관계의 표현

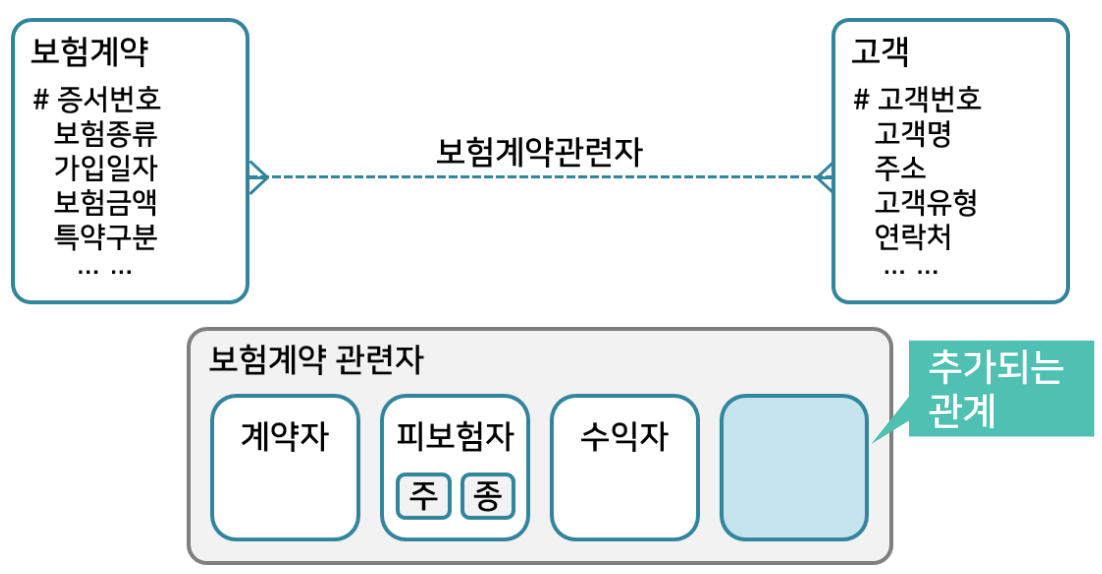
직렬식 관계

- 두 개체 사이에 존재하는 몇 개의 관계를 모아 **상위 개념으로 통합**함으로써 **하나의 관계로 관리**하는 방법

» 예 | 병렬로 정의된 여러 관계를 모두 묶어 '보험계약 관련자'라는 명칭의 관계로 통합

- 여러 개의 M:1 관계를 통합하면 당연히 M:M 관계가 됨
→ 이 관계를 해소하면 '보험계약 관련자'라는 관계 개체가 탄생
- 이때 각각의 관계들이 서로 섞이지 않도록 하기 위하여 서브타입을 사용

» 예 | 병렬로 정의된 여러 관계를 모두 묶어 '보험계약 관련자'라는 명칭의 관계로 통합





바커 표기법

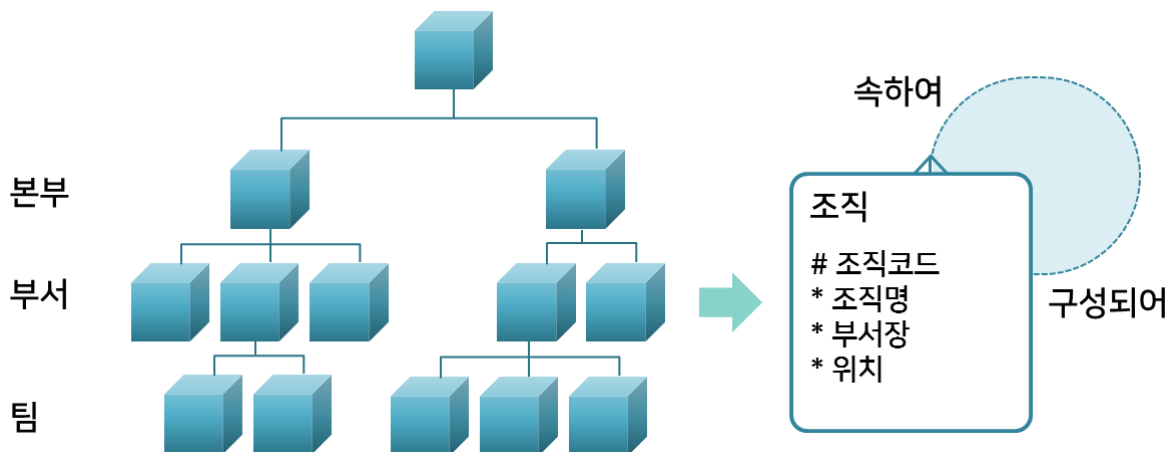


바커 표기법

10 관계의 표현

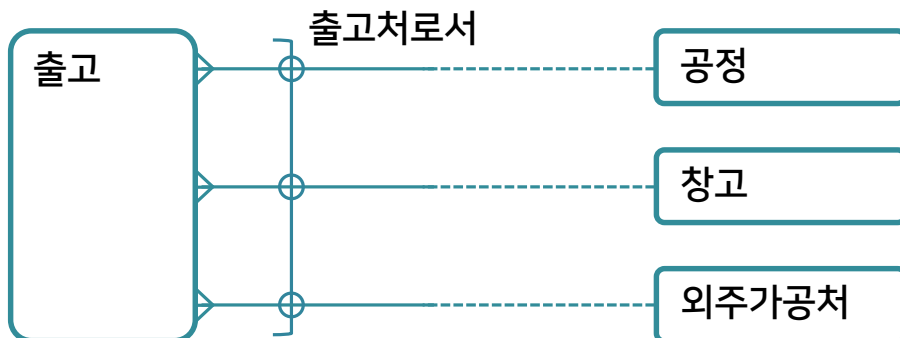
순환 관계(Recursive Relationship)

- 하나의 개체가 다른 개체가 아닌 자기 자신과 관계를 맺는 관계



아크 관계

- 어떤 개체가 두 개 이상의 다른 개체의 합집합과 관계를 가지는 것 (배타적(Exclusive) 관계 혹은 아크(Arc) 관계라고 불림)
- 동일한 의미의 관계가 서로 다른 하나 이상의 개체와 배타적으로 관계를 갖고 있을 때 이를 하나로 통합함으로써 발생





바커 표기법

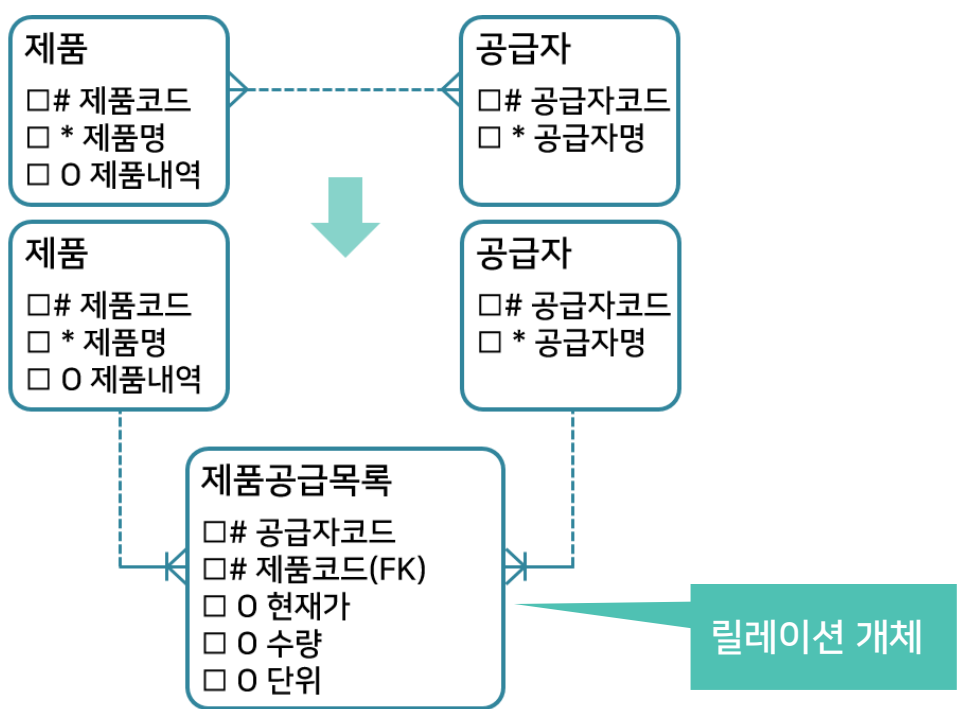


바커 표기법

11

M:M 관계의 해소

- 의외로 아주 빈번하게 발생하는 관계
 - 데이터 모델링이 완료된 후에 M:M은 존재하지 않음
 - M:M 관계가 아직 덜 풀려진 형태라는 것을 의미
- 관계가 해소되면서 새로운 개체(Relation Entity/Intersection Entity)를 생성





바커 표기법



바커 표기법

12

해시 마크

해당 개체 타입에서 **하나 이상의 개체가 관계에 참여**한다는 의미



▶ 실선이므로 같은 의미

▶ 모두 점선이므로 부분 참여의 의미

– 앞의 경우

B 타입 개체 중 관계에 참여하는 개체가 최소한 하나 이상은 있어야 한다는 의미

– 뒤의 경우

해시 마크가 없음으로 B 타입 개체 중 관계에 참여하는 개체 집합의 공집합이여도 된다는 의미



IE 표기법



IE 표기법

01

IE 표기법의 개념

IE 표기법의 탄생

- 1981년: Clive Finkelstein과 James Martin이 공동 저술로 발표
- 1980년대 중반: James Martin에 의해 그 체계가 정리되면서 본격적으로 활용

특징

- 정보시스템을 구축하는데 있어서 데이터 분석(Data Analysis)과 데이터베이스 설계(Database Design)를 위한 **매우 유용한 기법**
- 관계의 다(Many) 쪽을 나타내기 위해 까마귀 발을 사용하기 때문에 **까마귀 발 모델(Crow's Foot Model)**이라 불림



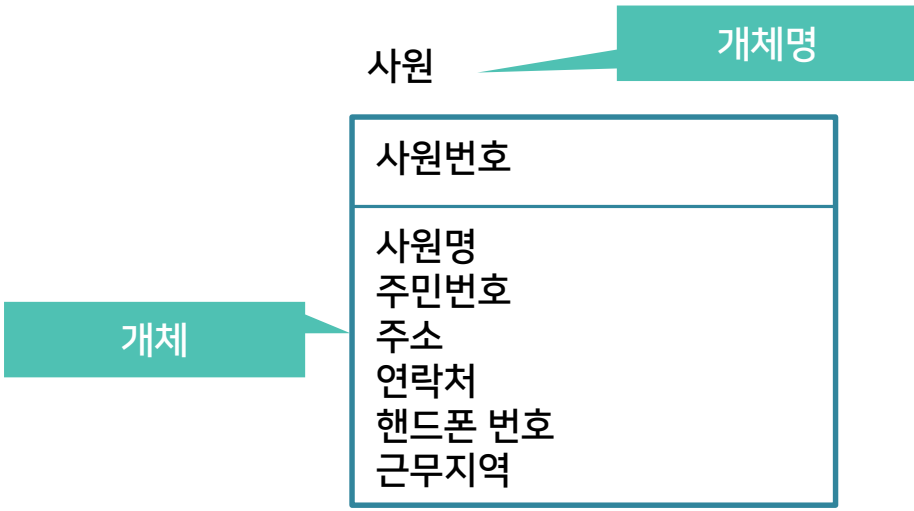
IE 표기법



IE 표기법

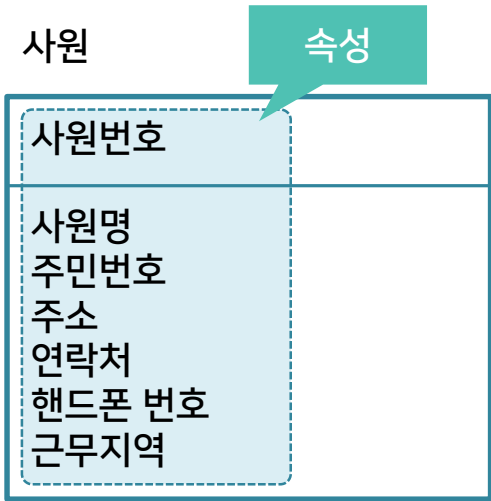
02 개체

- 사용자가 **추적**하고자 하는 어떤 사물



03 속성

- 개체의 특징을 기술해 주는 여러 개의 속성을 가짐
- 속성은 개체 안에 위치





IE 표기법



IE 표기법

04 식별자

- 개체는 그들을 지칭하거나 식별해 주는 속성인 식별자를 가지고 있음
- 속성의 식별자는 개체의 상단에 나타나며, 수평선이 식별자 밑에 그려짐

사원

사원번호
사원명 주민번호 주소 연락처 핸드폰 번호 근무지역

식별자



IE 표기법



IE 표기법

05

관계

- 까마귀 발 부호는 관계의 다(Many) 쪽을 보여주는 데 사용
- 타원(Oval), 해시 마크 및 까마귀 발의 다양한 조합들이 같이 사용됨

까마귀 발 부호의 사용	식별	비식별	의미
타원, 해시 마크 및 까마귀 발			0, 1 또는 그 이상의 개체 허용
까마귀 발이 있는 해시 마크			1 또는 그 이상의 개체 허용
해시 마크가 있는 타원			0 또는 1 개체 허용
해시 마크만 있음			정확히 1 개체 허용

06

식별 관계

실선 표시



식별은 관계를 통하여 이주한 부모의 **식별자가 자식의 주 식별자의 일부**가 되는 경우에 설정

점선 표시



비식별은 부모의 주 식별자가 자식의 **non-key 영역으로 이주**하고 자식을 식별하는 데 관계하지 않음



07

서브타입

배타적 또는 포괄적일 수 있음

만일 배타적이라면?

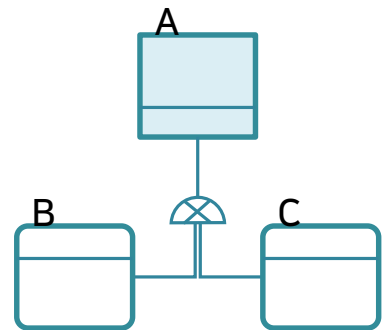
▶ 슈퍼타입은 많아야 **1개**의 서브타입과 관련될 수 있음

만일 포괄적이라면?

▶ 슈퍼타입은 **1개 또는 그 이상**의 서브타입과 관련될 수 있음

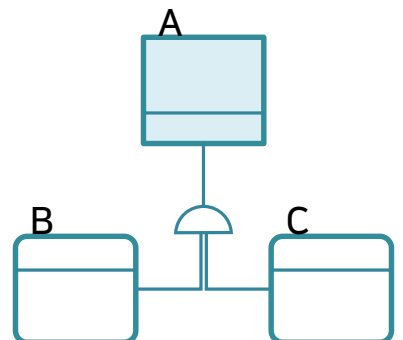
예시: A는 슈퍼타입, B와 C는 배타적 서브타입, 관계는 실선으로 그려져 있을 경우

» 예 | 직원은 반드시 일반직원과
촉탁직원 중 하나에만 속해야 함



예시: A는 슈퍼타입, B와 C는 포괄적 서브타입, 관계는 실선으로 그려져 있을 경우

» 예 | 접수 시 인터넷 접수를 한 사람은
다시 회사에 방문하여 방문 접수를
할 수 있음





IE 표기법



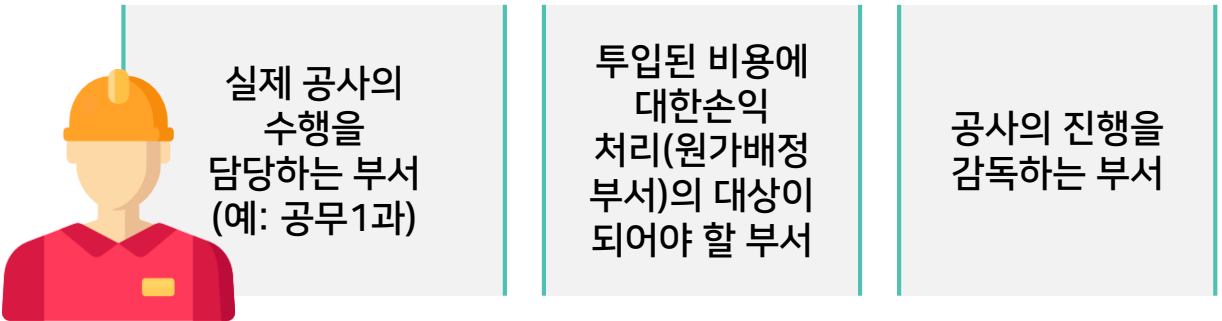
IE 표기법

08

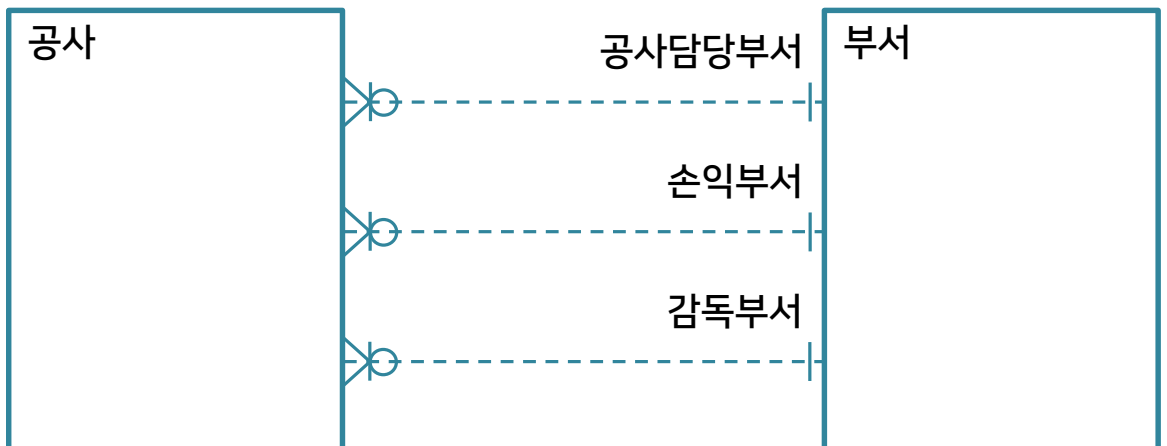
병렬식 관계

공사(工事) 개체의 의미

- 회사 내에서 일어나는 각종 토목이나 건축물을 구축하거나 보수하는 작업



예 | 생산1과가 관장하는 공정의 설비 개선 공사를 공무1과가 담당, 공무관리과에서 공사를 감독한다면?



손익 부서는 생산1과, 공사담당부서는 공무1과, 감독 부서는 공무관리과가 될 것임



IE 표기법

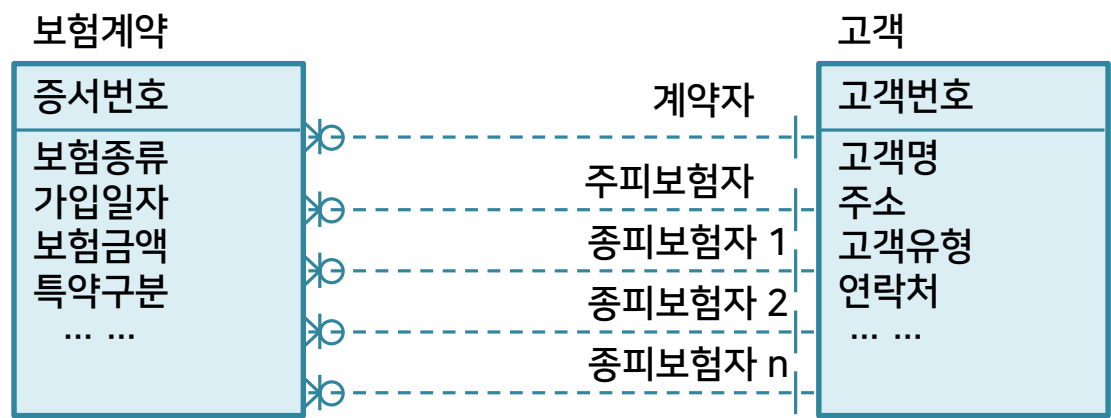


IE 표기법

09

직렬식 관계

- 병렬로 정의된 여러 관계를 모두 묶어 '보험계약 관련자'라는 명칭의 관계로 통합한 모양
- 여러 개의 M:1 관계를 통합하면 M:M 관계가 됨
 - 관계를 해소하면 보험계약 관련자란 관계 개체가 탄생
→ 각각의 관계들이 서로 섞이지 않도록 하기 위해 서브타입을 사용





IE 표기법

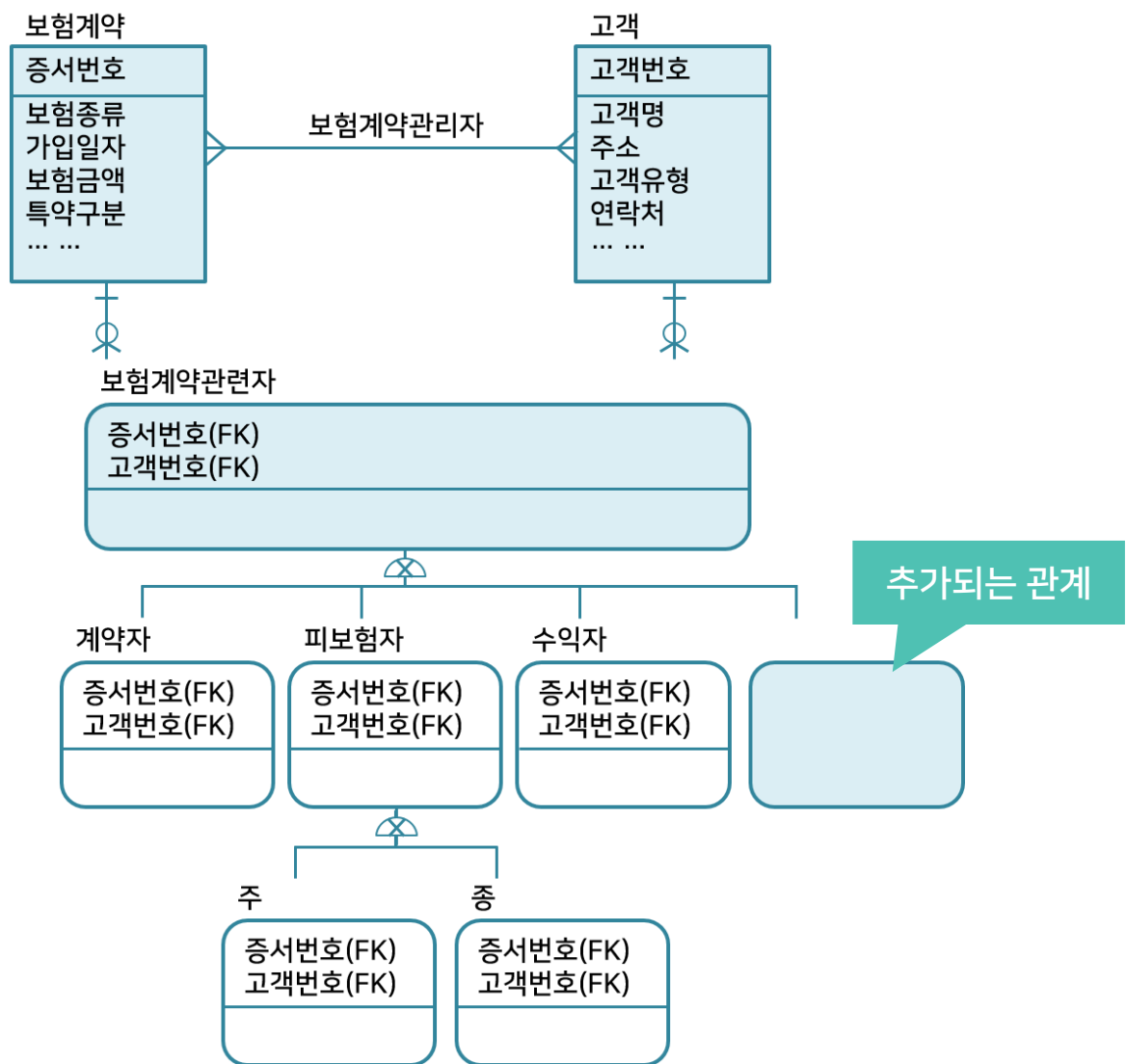


IE 표기법

09

직렬식 관계

- 관계를 해소하면 보험계약 관련자라는 **관계 개체**가 탄생
→ 각각의 관계들이 서로 섞이지 않도록 하기 위해 서브타입을 사용





개념 데이터 모델 작성

01 개념 데이터 모델의 구성 요소

- 핵심 개체(키 개체, 메인 개체)와 핵심 개체 사이의 관계 도출을 통해 핵심 구조라 할 수 있는 **데이터 모델의 골격을 정의**한 것

- 개념 데이터 모델의 생성

01

개괄 데이터 모델로부터 업무 요건을 충족하기 위해서 데이터 주제 영역 별로 상세화하여 핵심 개체들을 도출, 관계를 정의함으로써 생성

02

수집된 개체 후보들을 검토하고 분류하여 핵심 개체들을 도출하고 이들 간의 관계를 정의함으로써 생성

전사 개념
데이터 모델

- 주제 영역별로 작성된 개념 데이터 모델을 **전사 영역으로 확장하여 하나의 개념 데이터 모델로 작성**한 것
- 확장 개념 데이터 모델(Extended Conceptual Data Model)으로도 불림

전사 개념 데이터 모델(Enterprise Conceptual Data Model)은 전사 관점의 핵심 개체와 관계를 통해 **전사적인 핵심 데이터 구조를 정의**



개념 데이터 모델 작성

01 개념 데이터 모델의 구성 요소

개념 데이터 모델의 특징

- 01 시스템에 대한 사용자의 요구사항을 정형화된 모델로 표현
 - ▶ 사용자가 요구하는 데이터의 범위 및 구조를 용이하게 확인할 수 있음
- 02 사용자의 요구사항을 사용자와 함께 검토하여 신규 시스템에 해당 요구사항을 반영할지의 여부를 결정하여 개발 범위를 정하는데 도움을 줌
- 03 개괄 데이터 모델 및 전사 개념 데이터 모델과의 불일치가 발생하지 않도록 데이터의 골격을 유지
- 04 향후의 논리 및 물리 데이터 모델과의 구조적 정렬(Alignment)을 지원



개념 데이터 모델 작성

01

개념 데이터 모델의 구성 요소

개념 데이터 모델의 목적

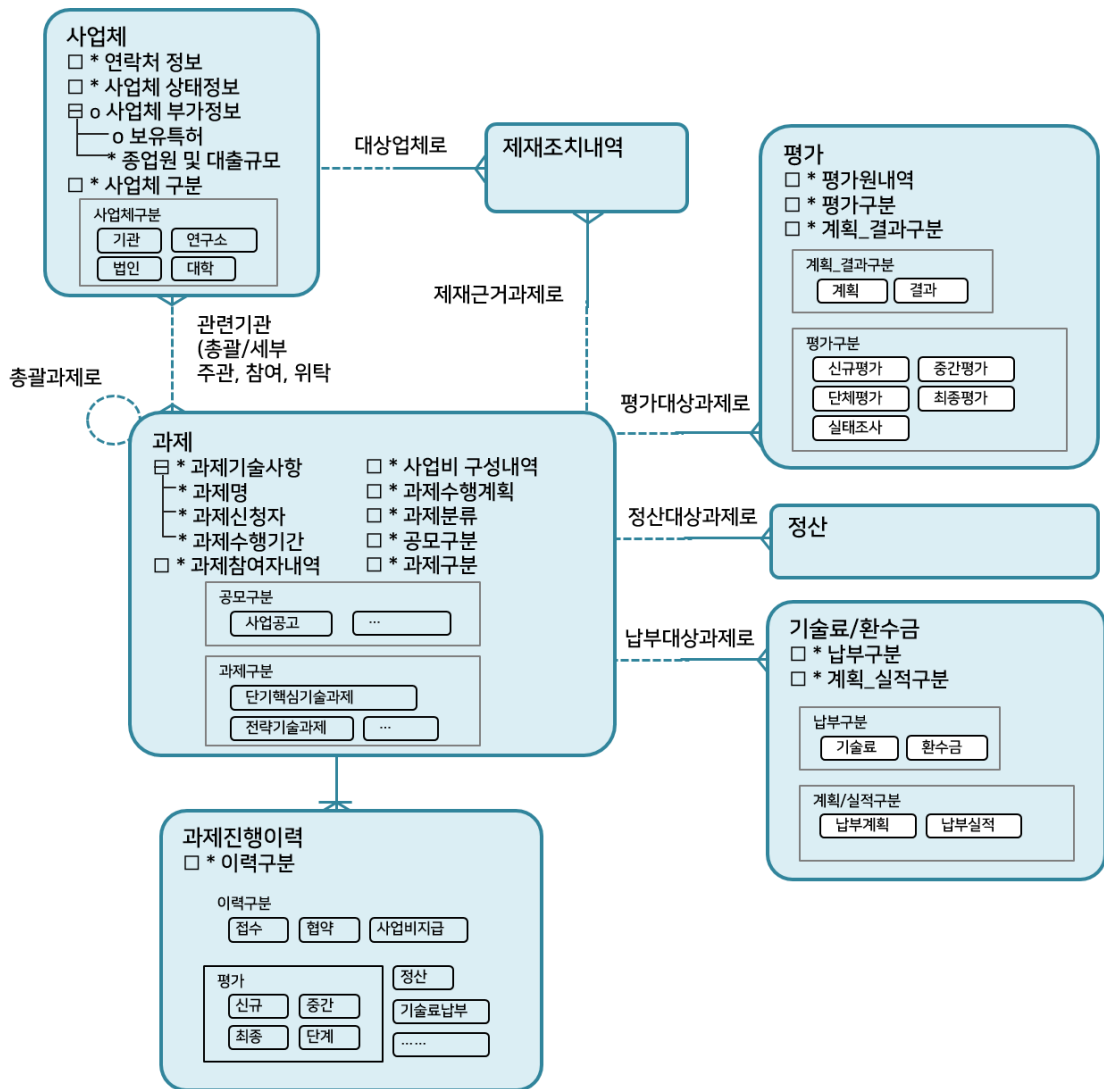
- **주요 핵심 개체와 이들 간의 관계 정의 위주로 구성하여 데이터의 전체적인 골격을 파악**
 - 작성된 개념 데이터 모델 상에서는 속성 표현이 불필요할 수 있으나, 개념 데이터 모델을 구성하는 핵심 개체들이 데이터의 발생 기원 측면에서 상위 수준에 해당
 - 전체 데이터 구성에 대한 이해를 돕기 위해 향후에 세분화되거나 자식으로 탄생될 수 있는 하위 개체나 부가적인 개체들을 속성이나 서브타입 형태로 표현하는 경우도 있음
- 개체를 이해하는데 도움을 주거나 해당 개체의 속성으로 **명백한 일부 핵심 속성을 함께 도출**



개념 데이터 모델 작성

02 개념 데이터 모델의 작성 예

바커 표기법

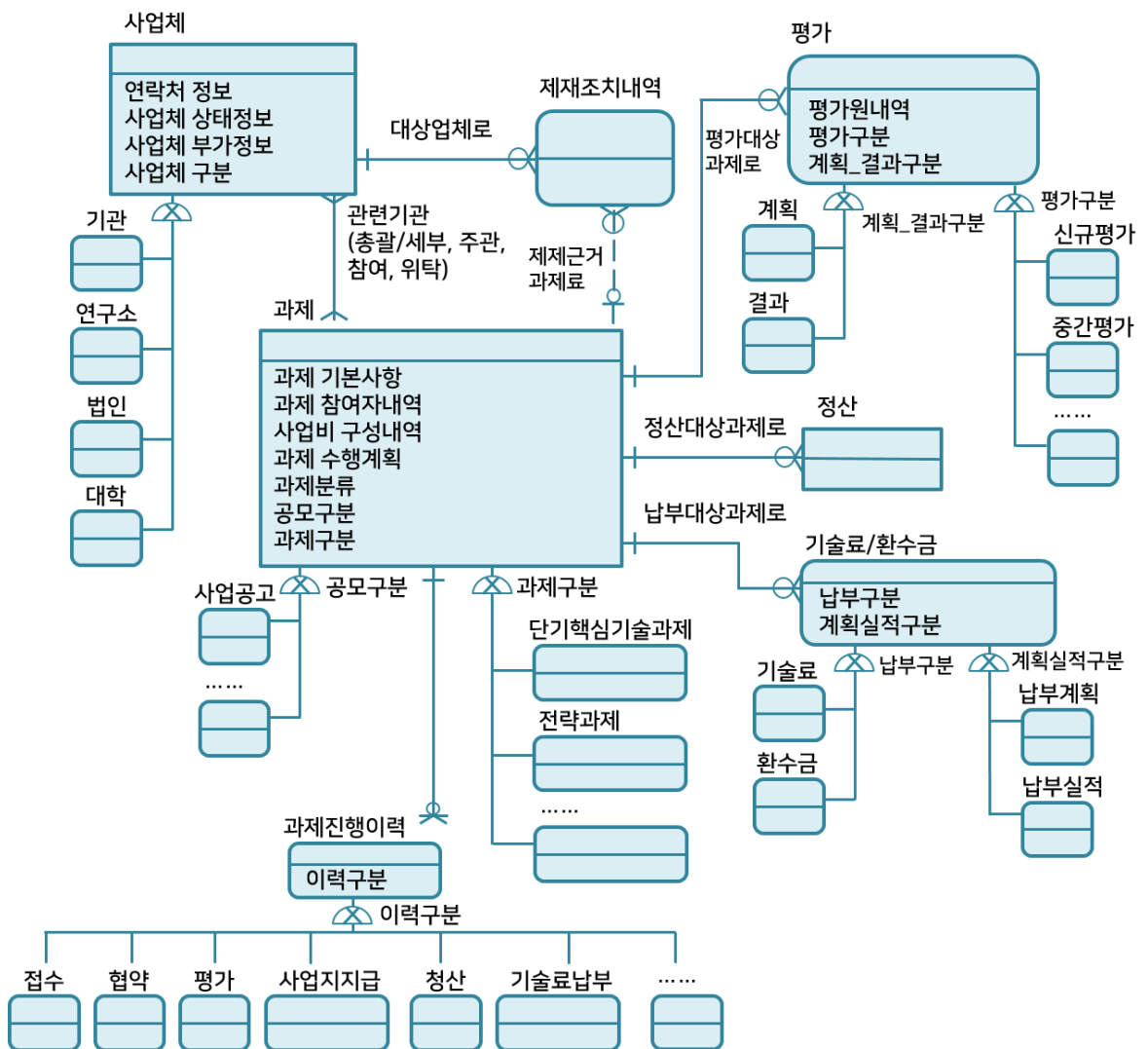




개념 데이터 모델 작성

02 개념 데이터 모델의 작성 예

IE 표기법





개념 데이터 모델 작성

03 개념 데이터 모델을 미작성 할 경우의 영향

01

개괄 데이터 모델 및 전사 개념 데이터 모델과의 불일치 사항이 발생할 수 있음

02

논리 및 물리 데이터 모델 작성 시 사용자 요구사항 반영이 누락되거나 잘못 반영될 수 있음

03

주제 영역 간 혹은 업무 간 데이터 연관에 있어 범위가 불명확해져 오류가 발생할 수 있음

04 개념 데이터 모델 작성이 불필요한 경우

- 개념 데이터 모델은 반드시 작성하는 것을 원칙으로 하는 것이 바람직함
- 하지만 시스템의 특성이 대체적으로 연산처리 중심이거나 업무에 연관된 데이터를 저장하지 않는 경우는 생략 가능함



개념 데이터 모델 작성

05 개괄 데이터 모델로부터 상세화 하는 경우

1

개괄 데이터 모델
또는 전사 개념
데이터
모델로부터 개념
데이터 모델을
작성할 단위 주제
영역을 결정

2

도출된 핵심
개체를 주제
영역과 매핑

3

주제 영역별로
해당 개체들 간의
관계를 정의하여
개념 데이터
모델을 작성

4

개념 데이터
모델을 상위
수준의
어플리케이션
모델이나
프로세스 모델과
비교하여 검증

06 수집된 후보 개체로부터 작성하는 경우

1

수집된 개체
후보들을
검토하고
분류하여 핵심
개체 도출

2

수집된 개체
후보들과 핵심
개체들을
분류하여
데이터 주제
영역 정의

3

데이터 주제
영역별로
해당하는 핵심
개체들을
배치하고 이들
간의 관계를
정의하여 개념
데이터 모델 작성

4

개념 데이터 모델을
상위 수준의
어플리케이션
모델이나
프로세스 모델과
비교하여 검증



개념 데이터 모델 작성

07 현행 데이터 리버스를 통해 작성하는 경우

1

현행 물리 데이터 모델을 생성하고 상세화 및 논리화를 거쳐 현행 논리 데이터 모델을 작성

2

현행 논리 데이터 모델의 개체들을 분류하여 핵심 개체를 도출하고 현행 데이터 주제 영역에 매핑

3

현행 논리 데이터 모델로부터 현행 핵심 개체 간의 관계를 정의하여 현행 개념 데이터 모델을 작성

4

사용자 요구사항이나 현행 시스템 분석 결과 또는 선진 사례 등을 검토하여 개선 사항을 반영함으로써 현행 개념 데이터 모델로부터 목표 개념 데이터 모델을 생성

5

목표 개념 데이터 모델을 상위 수준의 어플리케이션 모델이나 프로세스 모델과 비교하여 검증



1 바커 표기법

- ✓ 개체: 둥근 사각형
- ✓ 필수 속성: *
- ✓ 선택 속성: o
- ✓ 식별자: #
- ✓ 필수 참여: 실선
- ✓ 부분 참여: 점선
- ✓ 다수 참여: 까마귀 발

2 IE 표기법

- ✓ 개체: 사각형
- ✓ 식별자: 수평선
- ✓ 부분참여: 타원
- ✓ 다수참여: 까마귀 발