

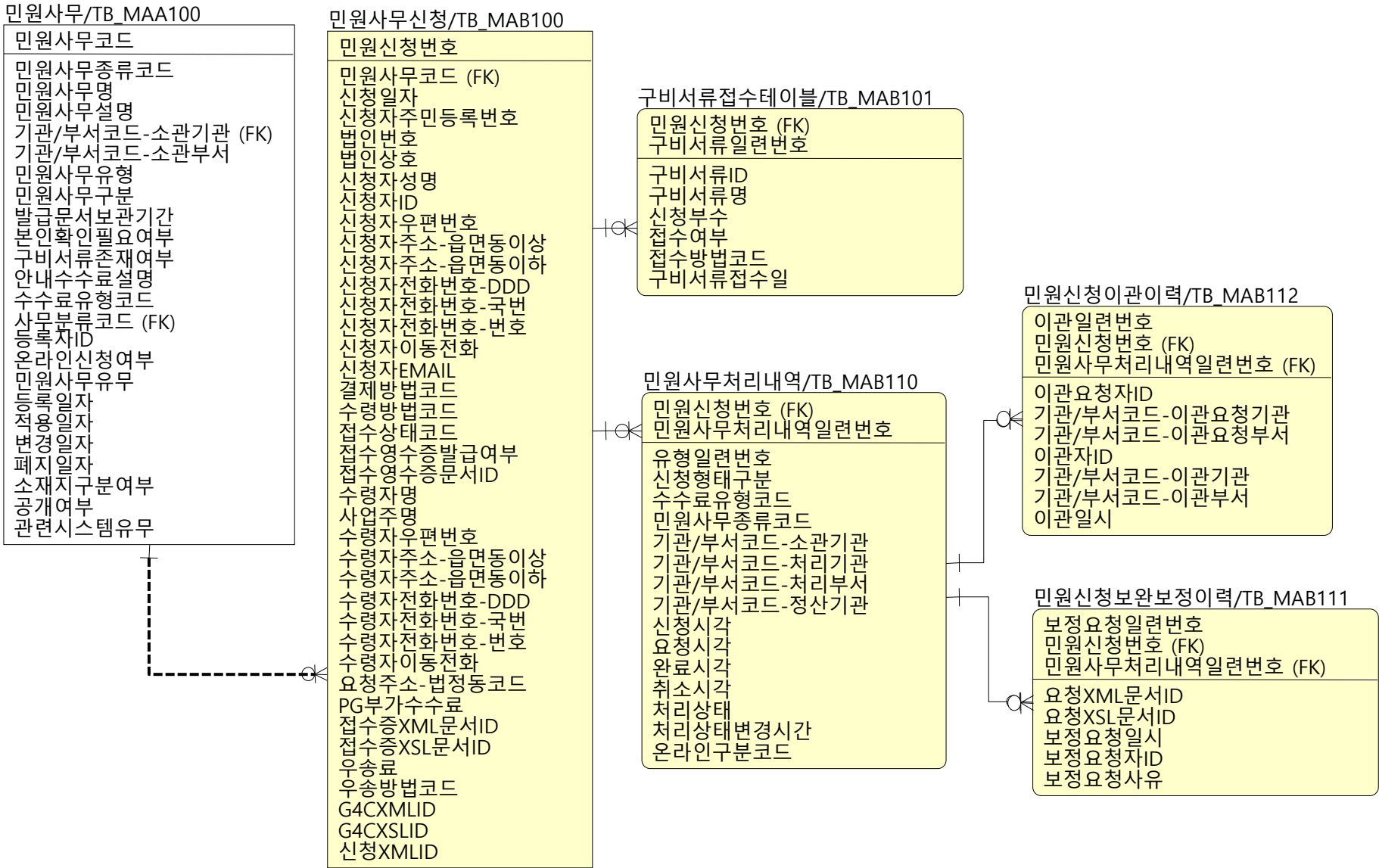
데이터모델링 특강

- 2020년 공공 빅데이터 청년 인턴십 데이터 전문교육과정 -

DB설계, C. S. Lee

궁금하면 질문하고 같이 토론해 봐요^^

민원사무/TA_MAA100 민원사무신청/TA MAB100



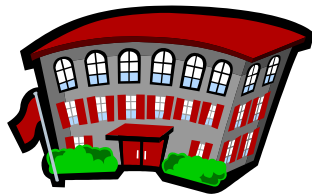


1. 엔터티
2. 속성
3. 관계
4. 식별자

■ 엔터티의 정의

- Peter Chen : 유일하게 식별될 수 있는 사물 (1976)
- C.J.Date : 데이터베이스내에 나타내는 구별할 수 있는 객체 (1986)
- James Martin : 우리가 그에 대한 정보를 보관하는 어떤 것이나 가능하다. (1989)
- Thomas Bruce : 정보로 관리되어야 하는 식별 가능한 사람, 장소, 사물, 사건, 개념 (1992)

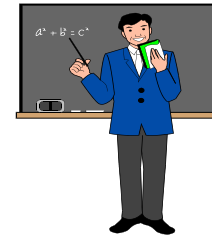
■ 엔터티의 정의



강의실



과목



강사

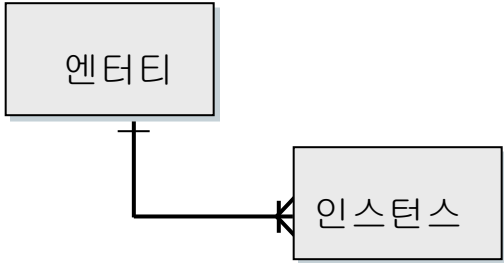
엔터티는 업무에 필요하고 유용한 정보를 저장하고
관리하기 위한 것으로 영속적으로 존재하는 단위

* 엔터티는 엔터티 타입 또는 엔터티 클래스 라고도 함

■ 엔터티 - 인스턴스

- ✓ 엔터티를 표현하는 방법은 각각의 표기법에 따라 조금씩 차이는 있지만 대부분 사각형으로 표현된다. 다만 이 안에 표현되는 속성의 표현방법이 조금씩 다를 뿐이다. 엔터티와 엔터티간의 ERD를 그리면 그림과 같이 표현할 수 있다.

엔터티-인스턴스 ERD



엔터티는 인스턴스의 집합

엔티티-인스턴스의 예

엔터티	인스턴스
과 목	수학
	영어
강 사	이춘식
	이현철
사 건	2010-001
	2010-002

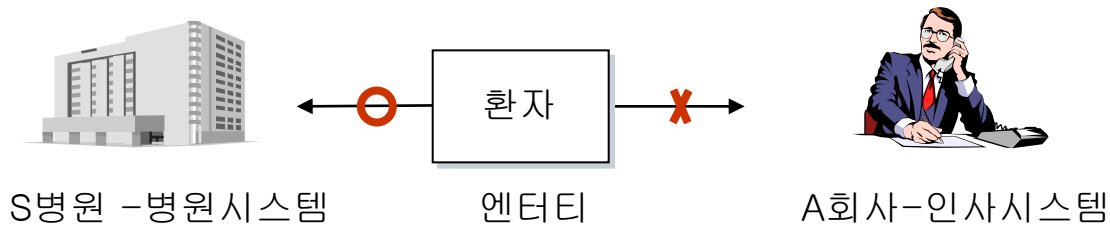
- 엔터티의 특징

- ✓ 반드시 해당 업무에서 필요하고 관리하고자 하는 정보이어야 한다.
- ✓ 유일한 식별자에 의해 식별이 가능해야 한다.
- ✓ 영속적으로 존재하는 인스턴스의 집합이어야 한다.('한 개'가 아니라 '두 개 이상')
- ✓ 엔터티는 업무 프로세스에 의해 이용되어야 한다.
- ✓ 엔터티는 반드시 속성이 있어야 한다.
- ✓ 엔터티는 다른 엔터티와 최소 한 개 이상의 관계가 있어야 한다.

- 엔터티특징 - 업무에서 필요로 하는 정보

✓ 반드시 시스템을 구축하고자 하는 업무에서 필요로 하고 관리하고자 하는 정보여야 한다는 점이다.

업무에서 관리할 필요가 있는가?



업무에서 관리하고자 하는 영역(Business Boundary)에 대한 인식이 매우 중요함

- 엔터티특징 - 식별이 가능해야 함

- ✓ 유일한 식별자는 그 엔터티의 인스턴스만의 고유한 이름이다. 두 개 이상의 엔터티를 대변하면 그 식별자는 잘못 설계된 것이다.

유일한 식별자를 가질 수 있는가?



모두다 동일한 이름, 속성, 관계??

인스턴스 각각을 구분하기 위한 유일한 식별자가 존재해야 함

• 엔터티특징 – 인스턴스 집합

- ✓ 엔터티의 특징 중 “한 개”가 아니라 “두 개 이상”이라는 집합개념은 매우 중요한 개념

엔터티는 두 개 이상의 인스턴스를 가지는가?

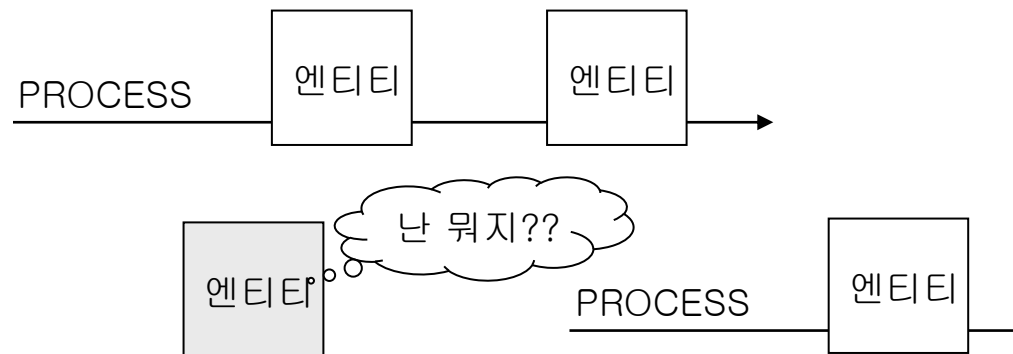


인스턴스가 한 개 밖에 없는 회사, 병원 엔터티는 집합이 아니므로 엔터티 성립이 안됨

- 엔터티특징 - 업무프로세스에 의해 이용

- ✓ 업무프로세스에 의해 전혀 이용되지 않는다면 업무 분석이 정확하게 안되어 엔터티가 잘못 선정되거나 업무프로세스 도출이 적절하게 이루어지지 않았음을 의미한다

엔터티는 프로세스에 의해 이용되는가?



업무 프로세스에 의해 이용되지 않는 엔터티는 그 업무의 엔터티가 아님

• 엔터티특징 - 속성포함

- ✓ 업무프로세스에 의해 전혀 이용되지 않는다면 업무 분석이 정확하게 안되어 엔터티가 잘못 선정되거나 업무프로세스 도출이 적절하게 이루어지지 않았음을 의미한다
- ✓ 예외적으로 관계엔터티(Associative Entity)의 경우는 주식별자 속성만 가지고 있어도 엔터티로 인정한다.

엔터티는 속성을 포함하는가?

태풍
태풍이름
발생지역
중속

날씨
날씨이름

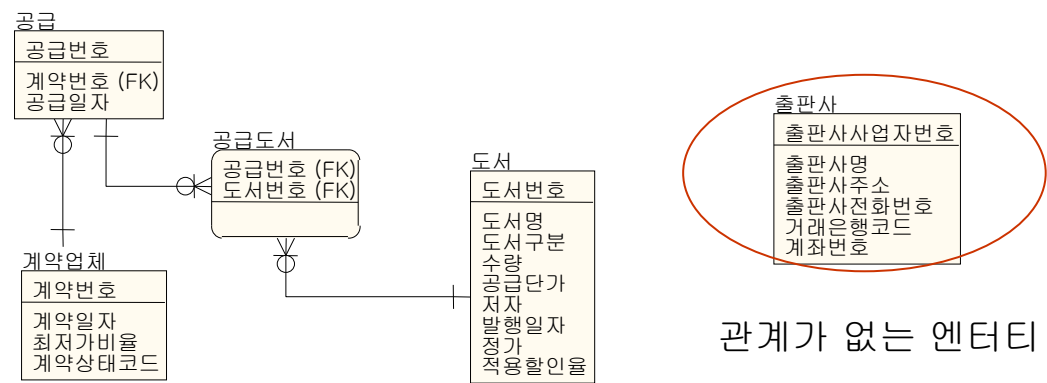
엔티티?

속성이 존재하지 않는 오브젝트는 엔터티가 될 수 없음

엔터티특징 - 관계의존재

- ✓ 다른 엔터티와 최소 한 개 이상의 관계가 존재해야 한다는 것이다.
- ✓ 기본적으로 엔터티가 도출되었다는 것은 해당 업무내에서 업무적인 연관성(존재적 연관성, 행위적 연관성)을 가지고 다른 엔터티와의 연관의 의미를 가지고 있음을 나타낸다.
- ✓ 단, 데이터 모델링을 하면서 관계를 생략하여 표현해야 하는 경우는 다음과 같은 통계성 엔터티 도출, 코드성 엔터티 도출, 시스템 처리시 내부 필요에 의한 엔터티 도출과 같은 경우이다.

엔터티는 관계를 가지고 있는가?

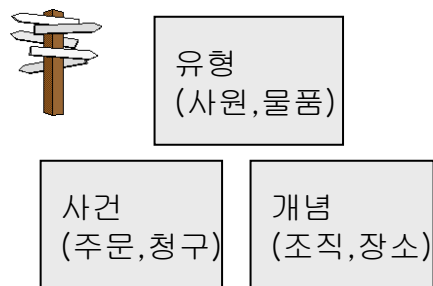


엔터티가 관계가 없으면, 잘못된 엔터티이거나 관계가 누락되었을 가능성이 큼

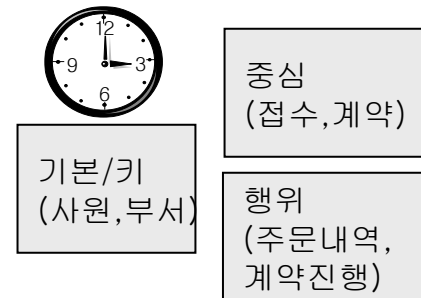
- 엔터티의 분류

- ✓ 엔터티는 엔터티 자신의 성격에 의해 실체유형에 따라 구분하거나 업무를 구성하는 모습에 따라 구분이 되는 발생시점에 의해 분류할 수 있다.

유무형에 따라...



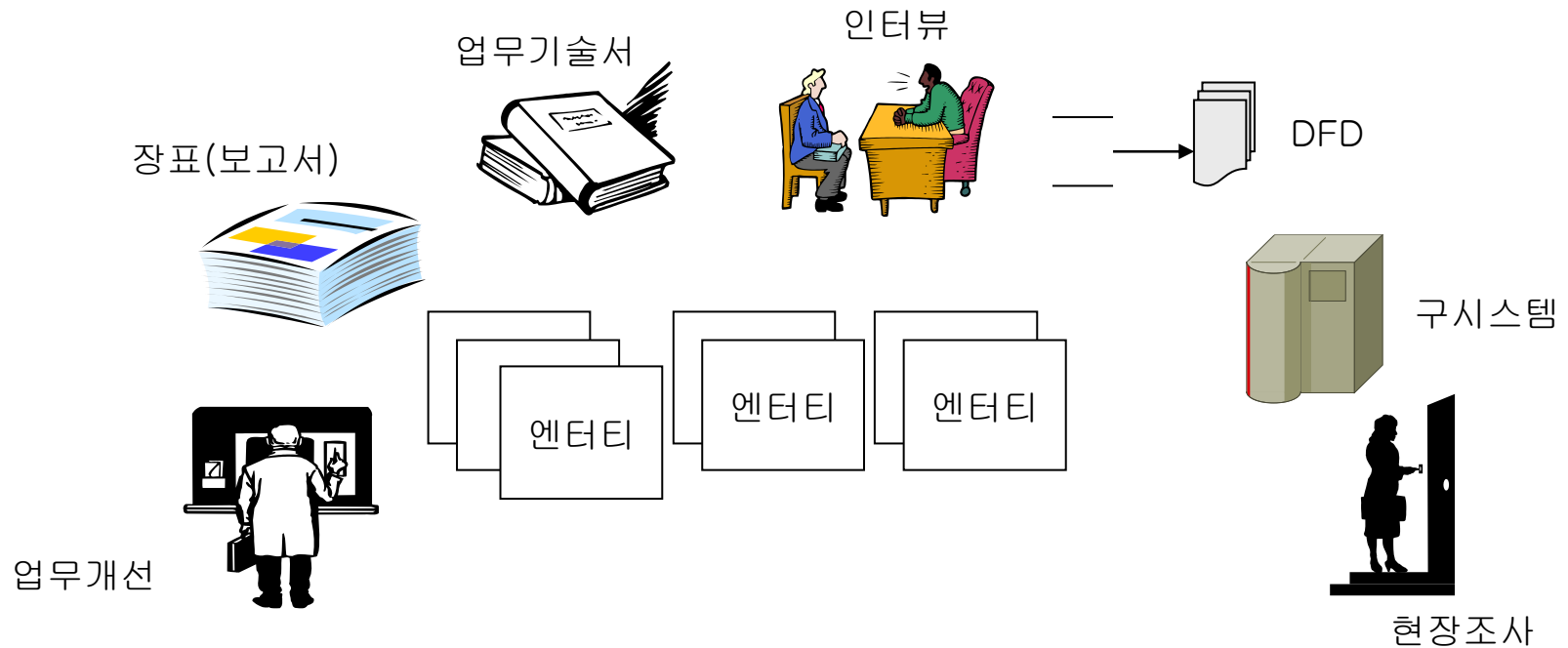
발생시점에 따라...



엔터티를 도출할 때 일정한 그룹에 의해 그룹화하면 도출작업에 효율적임

- 엔터티의 명명

- ✓ 첫 번째는 가능하면 현업업무에서 사용하는 용어를 사용한다.
- ✓ 두 번째는 가능하면 약어를 사용하지 않는다.
- ✓ 세 번째는 단수명사를 사용한다.
- ✓ 네 번째는 모든 엔터티에서 유일하게 이름이 부여되어야 한다.
- ✓ 다섯 번째는 엔터티 생성의미대로 이름을 부여한다.



엔터티를 찾아내는 작업은 검사가 범죄자의 증거자료를 찾듯이 가능한 모든 방법이 동원된다.

• 속성(Attribute)의 정의

- ✓ 속성이란 사전적인 의미로는 사물(事物)의 성질, 특징 또는 본질적인 성질, 그것이 없다면 실체를 생각할 수 없는 것으로 정의
- ✓ “업무에서 필요로 하는 인스턴스로 관리하고자 하는 의미상 더 이상 분리되지 않는 최소의 데이터 단위”
- ✓ 업무에서 필요로 한다.
- ✓ 의미상 더 이상 분리되지 않는다.
- ✓ 엔터티를 설명하고 인스턴스의 구성요소가 된다.

엔터티



엔터티는 속성들에 의해 설명된다.

속성

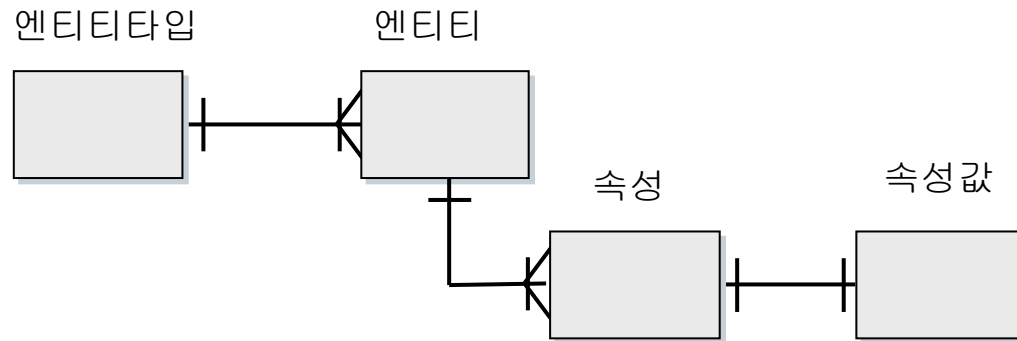
- 이름
- 주소
- 생년월일
- 계약일자
- 전문분야

인스턴스의 에서 관리하고자 하는 의미상 더 이상 분리되지 않는 최소의 데이터 단위

- 속성 - 엔터티, 인스턴스, 속성, 속성값의 관계

- ✓ 한 개의 엔터티는 두 개 이상의 인스턴스의 집합이어야 한다.
- ✓ 한 개의 엔터티는 두 개 이상의 속성을 갖는다.
- ✓ 한 개의 속성은 한 개의 속성값을 갖는다.

엔터티-인스턴스-속성-속성값 ERD



- 속성 - 속성표기법

- ✓ 속성의 표기법은 엔터티 내에 이름을 포함하여 표현하면 된다

과목	
과목이름	
교재이름 생성일자	

강사	
강사이름	
주소 생년월일	

사건	
사건번호	
발생장소 발생일시	

- 속성의 특징

- ✓ 엔터티와 마찬가지로 반드시 해당 업무에서 필요하고 관리하고자 하는 정보이어야 한다. (예, 강사의 교재이름)
- ✓ 정규화 이론에 근간하여 정해진 주식별자에 함수적 종속성을 가져야 한다.
- ✓ 하나의 속성에는 한 개의 값만을 가진다. 하나의 속성에 여러 개의 값이 있는 다중값일 경우 별도의 엔터티를 이용하여 분리한다.

- 속성의 분류 - 특성에 따른 분류

- ✓ 속성은 업무분석을 통해 바로 정의한 속성을 기본속성(Basic Attribute), 원래 업무상 존재하지는 않지만 설계를 하면서 도출해내는 속성을 설계속성(Designed Attribute), 다른 속성으로부터 계산이나 변형이 되어 생성되는 속성을 파생속성(Derived Attribute)이라고 한다.

기본(BASIC)-원래속성



제품이름
제조년월
제조원가

설계(DESIGNED) -1:1 치환

001-식품용기
002-약품용기
003-기타용기

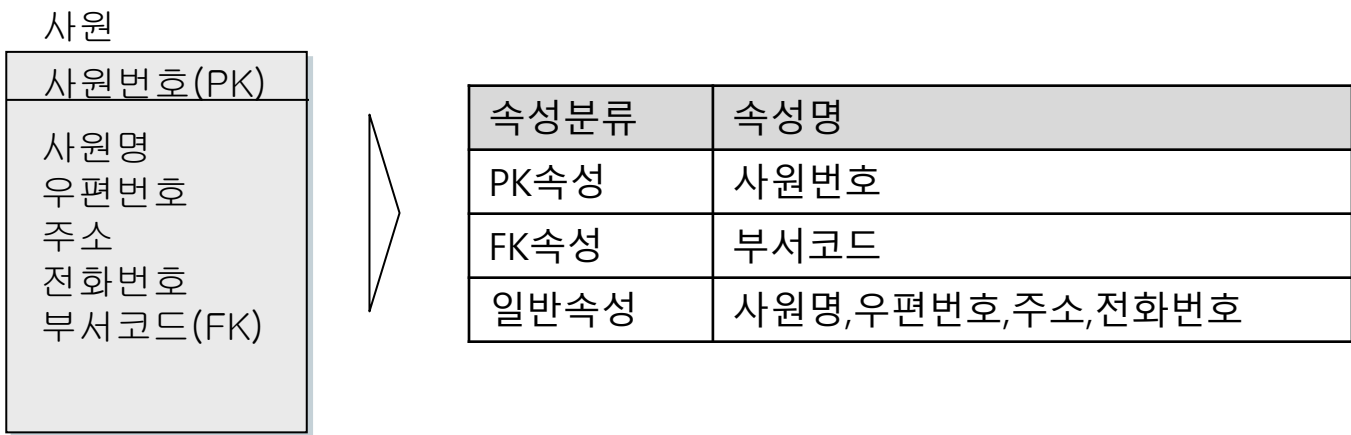
약품용기코드

파생(DERIVED) -계산값

전체용기 수(Σ 개별용기)
용기의 총금액(Σ 단가)

계산값

- 속성의 분류 - 엔터티 구성방식에 따른 분류
 - ✓ 엔터티를 식별할 수 있는 속성을 PK(Primary Key)속성, 다른 엔터티와의 관계에서 포함된 속성을 FK(Foreign Key)속성, 엔터티에 포함되어 있고 PK, FK에 포함되지 않은 속성을 일반속성이라 한다.

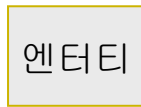


- 속성의 명명(Naming)
 - ✓ 해당업무에서 사용하는 이름을 부여 한다.
 - ✓ 서술식 속성명은 사용하지 않는다.
 - ✓ 약어사용은 가급적 제한한다.
 - ✓ 전체 데이터모델에서 유일성 확보하는 것이 좋다.

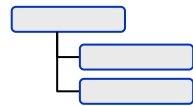
- 속성의 명명(Naming) 규칙 사례

1. 시스템 용어사전에 등록된 용어만을 사용하여 가능하면 5 단어 이내로 작성한다.
2. 속성명은 Qualifier + Domain 형태로 명명하며,
다른 속성명과 구분이 어려운 경우 엔터티 명을 추가적인 Qualifier로 사용한다.
예) 전화번호 → 부서(Qualifier) + 전화번호(Domain)
3. 반드시 도메인 정의서에서 정의된 관련 컬럼 접미어로 끝나도록 한다.
예) 금액, 양(량, 비율, 율, 료), 일자(연월, 시간, 개월, 기간), 명(성명),
코드(구분, 코드, 여부, 유무), 수(건수, 차수), ID(사번), 점수(점, 평점) 등
4. 논리 데이터 모델의 속성명은 한글로 기입함을 원칙으로 하고,
한글 명사 간에는 '/', '-', 띄어쓰기 등을 넣지 않는다.
5. 한글 적용이 불가능한 영문단어일 경우 영문을 사용할 수 있다.
영문은 대문자 알파벳만 사용하며 단어 간에는 띄어쓰기를 사용하지 않으며
단어 구분이 필요한 경우 '_'를 사용한다.
예) IP_ADDRESS

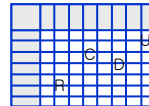
• 속성의 도출



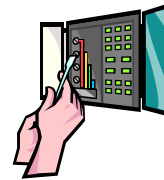
엔터티
선정



프로세스
모델링



상관모델링



현시스템
분석



업무자료를
수집하는 동안

속성을 발견하는 작업은 개발 이전까지 지속적으로 진행되어야 하므로
논리 데이터 모델링 단계에서 완벽하게 속성을 찾을 필요는 없음

- 속성의 도출

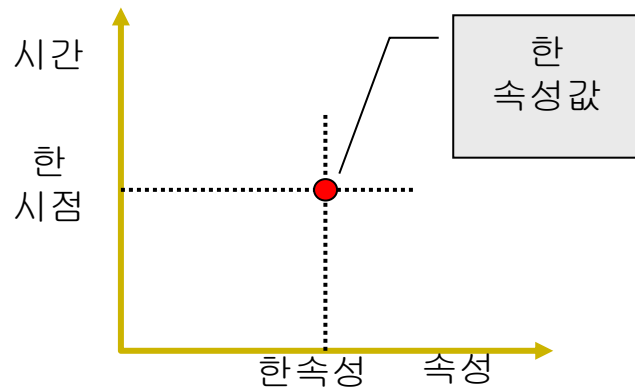
나는 어디로 가야하나???



업무적으로 관리해야 할 속성이 있음에도 불구하고 도출된 엔터티가 존재하지 않을 경우는 새로운 엔터티 생성을 고려해야 한다.

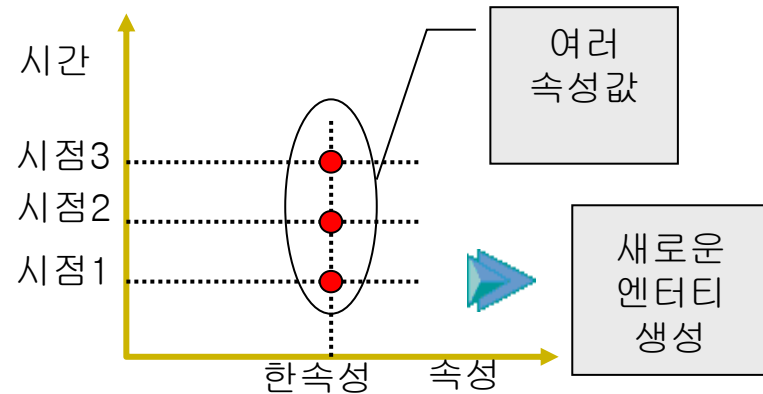
속성의 111원칙

속성 111 원칙



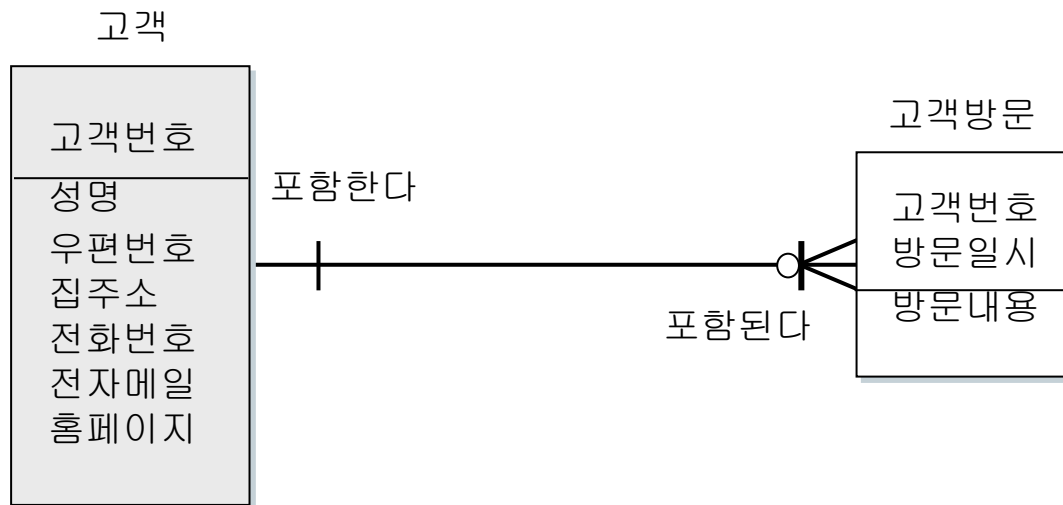
한 속성은 한 시점에 한 개의 값만을 가질 수 있다.

속성에 의한 엔터티 생성



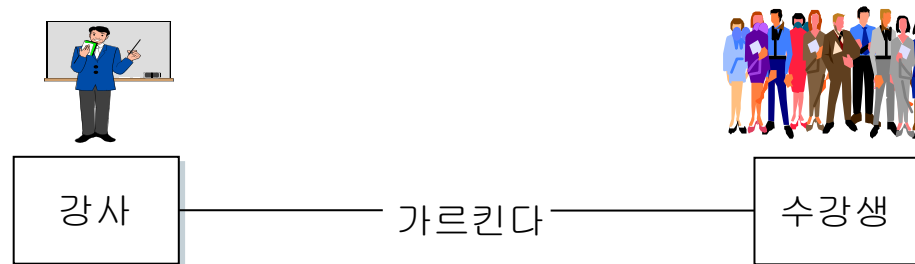
하나의 속성이 시간에 따라 여러 개의 값을 가지며 그 값을 해당업무에서 관리해야 할 필요가 있으면 엔터티를 생성해서 관리한다.

- 속성으로 인한 엔터티 도출



- 관계의 정의

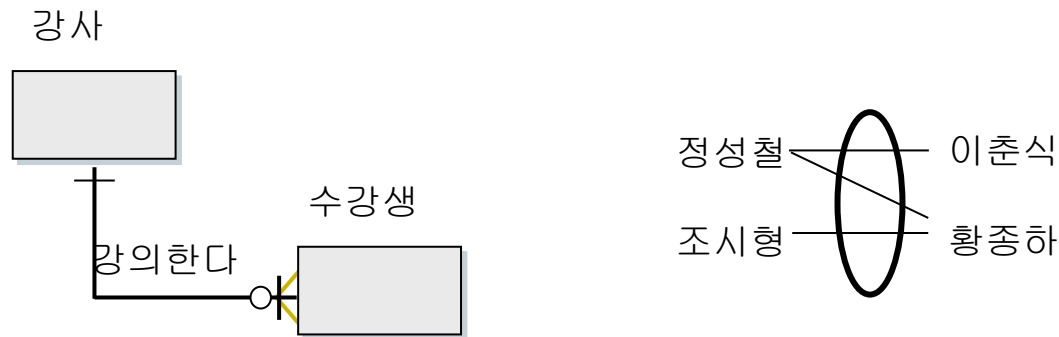
- ✓ 사전적으로 정의하면 상호 연관성이 있는 상태이다
- ✓ “엔터티의 인스턴스 사이의 논리적인 연관성으로서 존재의 형태로서나 행위로서 서로에게 연관성이 부여된 상태”라고 할 수 있다.



인스턴스 사이의 논리적인 연관성으로서 존재 또는 행위로서 서로에게 연관성이 부여된 상태

• 관계의 패어링

- ✓ 각각의 엔터티의 인스턴스들은 자신이 관련된 인스턴스들과 관계의 어커런스로 참여하는 형태

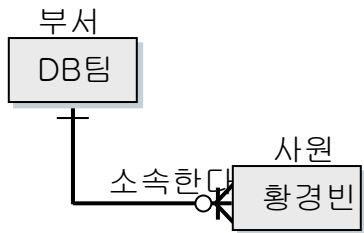


인스턴스 각각은 자신의 연관성을 가지고 있을 수 있음. 이것을 집합하여 '강의'라는 관계도출

- 관계의 분류

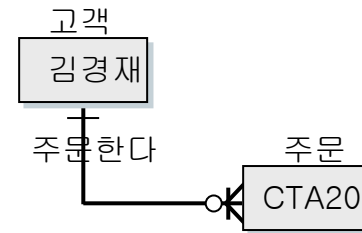
- ✓ 관계가 존재에 의한 관계와 행위에 의한 관계로 구분될 수 있는 것은 관계를 연결함에 있어 어떤 목적으로 연결되었느냐에 따라 분류하기 때문이다.

존재에 의한 관계



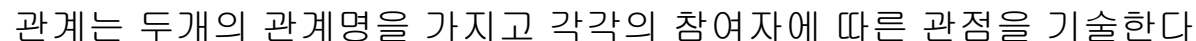
사원에 부서에 항상 속해있다

행위에 의한 관계



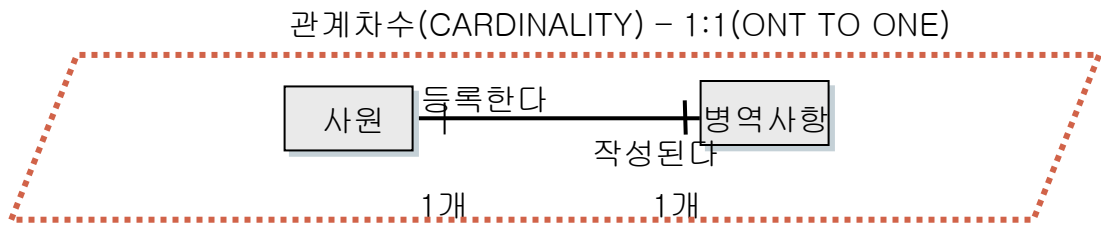
주문은 고객이 주문을 할 때 발생이 된다

- ✓ 엔터티에서 관계가 시작되는 편을 관계시작점(The Beginning)이라고 부르고 받는 편을 관계끝점(The End)이라고 부른다. 관계 시작점과 끝점 모두 관계이름을 가져야 하며 참여자의 관점에 따라 관계이름이 능동적(Active)이거나 수동적(Passive)으로 명명된다. 관계명은 다음과 같은 명명규칙에 따라 작성해야 한다.
- ✓ 애매한 동사를 피한다. 예를 들면 ‘관계된다’, ‘관련이 있다’, ‘이다’, ‘한다’ 등은 구체적이지 않아 어떤 행위가 있는지 또는 두 참여자간 어떤 상태가 존재하는지 파악할 수 없다.
- ✓ 현재형으로 표현한다. 예를 들면 ‘수강을 신청했다’, ‘강의를 할 것이다’라는 식으로 표현해서는 안 된다. ‘수강 신청한다’, ‘강의를 한다’로 표현해야 한다.



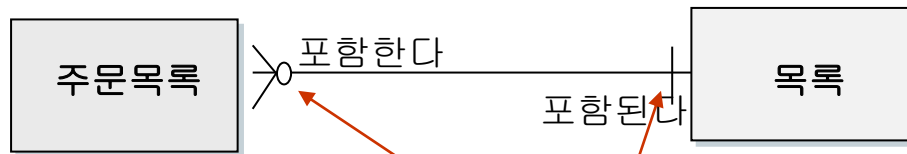
• 관계차수(Degree/Cardinality)

- ✓ 두 개의 엔터티간 관계에서 참여자의 수를 표현하는 것을 관계차수(Cardinality)라고 한다. 가장 일반적인 관계차수 표현방법은 1:M, 1:1, M:N이다. 가장 중요하게 고려해야 할 사항은 한 개의 관계가 존재하느냐 아니면 두 개 이상의 멤버십이 존재하는지를 파악하는 것이 중요하다.
- ✓ 관계차수를 표시하는 방법은 여러 가지 방법이 있지만 Crow's Foot 모델에서는 선을 이용하여 표현한다. 한 개가 참여하는 경우는 실선을 그대로 유지하고 다수가 참여한 경우는(Many) 까마귀발과 같은 모양으로 그려준다.



- 관계 선택사양

- ✓ 참여하는 엔터티가 항상 참여하는지 아니면 참여할 수도 있는지를 나타내는 방법이 필수 (Mandatory Membership)와 선택참여(Optional Membership)이다.
- ✓ 선택참여관계는 ERD에서 관계를 나타내는 선에서 선택참여하는 엔터티 쪽을 원으로 표시한다. 필수참여는 아무런 표시를 하지 않는다.



하나의 주문목록에는 한 개의 목록을 **항상** 포함하고
한 목록은 여러 개의 주문목록에 의해 **포함될 수** 있다.

- 관계의 정의 - 체크사항

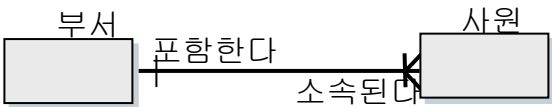
- ✓ 두 개의 엔터티 사이에 관심있는 연관규칙이 존재하는가?
- ✓ 두 개의 엔터티 사이에 정보의 조합이 발생하는가?
- ✓ 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?
- ✓ 업무기술서, 장표에 관계연결을 가능하게 하는 동사(Verb)가 있는가?

- 관계의 정의 - 체크사항

- ✓ 두 개의 엔터티 사이에 관심있는 연관규칙이 존재하는가?
- ✓ 두 개의 엔터티 사이에 정보의 조합이 발생하는가?
- ✓ 업무기술서, 장표에 관계연결에 대한 규칙이 서술되어 있는가?
- ✓ 업무기술서, 장표에 관계연결을 가능하게 하는 동사(Verb)가 있는가?

• 관계읽기

- ✓ 기준(Source) 엔터티를 한 개(One) 또는 각(Each)으로 읽는다.
- ✓ 대상(Target) 엔터티의 관계참여도 즉 개수(하나, 하나 이상)를 읽는다.
- ✓ 관계선택사양과 관계명을 읽는다.



각각의 / 하나의	기준엔터티 (Source)	관계 차수	관련엔터티 (Target)	선택사양 필수/ 선택	관계명
각각의	사원은	한	부서에	때때로	속한다
각	부서에는	여러	사원이	항상	소속된다

- 관계도출절차

1. 업무기술서, 장표, 인터뷰 정리문서 등에서 동사를 구분한다.
2. 도출된 엔터티와 관계를 이용하여 관계 정의서를 작성하도록 한다.
3. 고객에게 질문하여 관계에 대해 더 세분화하고 정확하게 도출하는 작업을 한다.
4. 데이터 모델링 도구(Tool)나 칠판, 포스트잇을 이용하여 직접 모델을 그려본다.
5. 고객에게 질문하고 협의하며 모델을 검토하는 시간을 갖는다.



- 관계도출절차

1. 업무기술서, 장표, 인터뷰 정리문서 등에서 동사를 구분한다.

엔터티가 명사형에 의해 구분되듯이 관계는 동사에 의해 구분된다.

강사는 여러 개의 강좌를 강의할 수 있다. 기술대학원에서는 여러 명의 강사를 기록하고 관리한다. 기술대학원에서 과목 당 개설한 강좌는 한 명의 강사가 강의를 진행한다.

☞ 관계의 대상은 ‘강의한다’, ‘기록 관리한다’, ‘개설한다.’ 이다.

• 관계도출절차

2. 도출된 엔터티의 관계를 이용하여 관계 정의서를 작성하도록 한다.

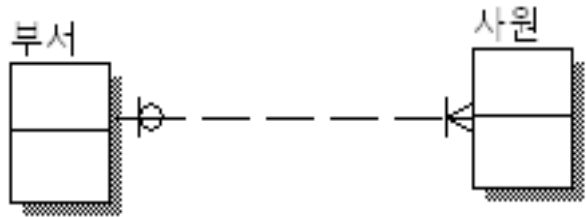
기준 엔터티	관계형태(방향,참여도,참여방법)	참여방법	관련 엔터티
사원	각각의 사원은 한 부서에 속한다	필수	부서
	각 부서에는 여러 명이 사원이 존재할 수 있다	선택	
	각각의 사원은 여러 개의 주문을 접수할 수 있다	선택	주문
	각각의 주문은 한명의 사원에 의해서만 접수된다	필수	

• 관계도출절차

3. 업무분석대상에게 질문하여 관계를 더 세분화하고 정확하게 정의한다.

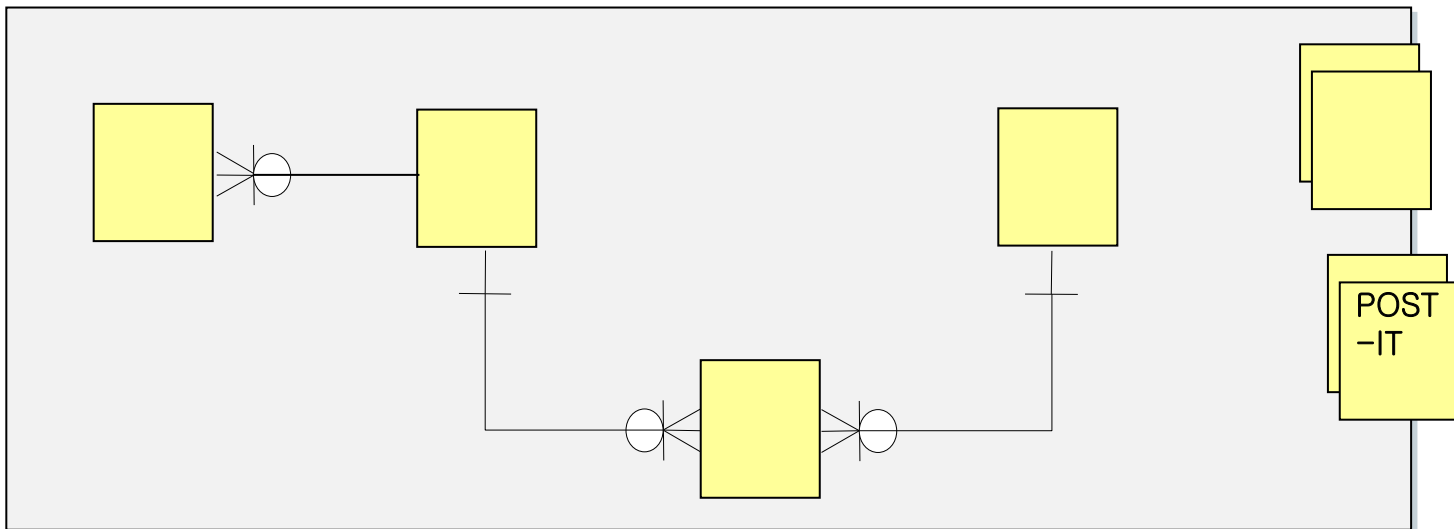
관계(Relationship) 선정을 위한 질문방법

각각의	기준엔터티	차수	관련엔터티	필수	참여방법
각각의	사원은	한	부서에	때때로	속할 수 있습니까?
각	부서에는	여러	사원이	항상	소속되어 있습니까?



- 관계도출절차

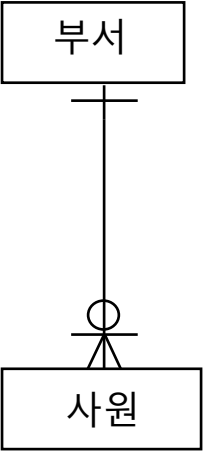
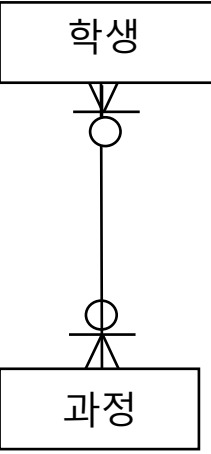
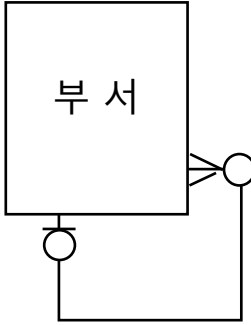
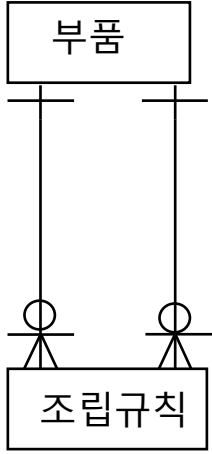
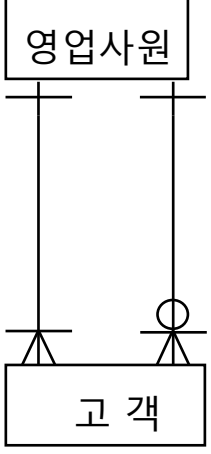
- 데이터 모델링 도구(Tool)나 칠판, 포스트잇을 이용하여 직접 모델을 그려본다.



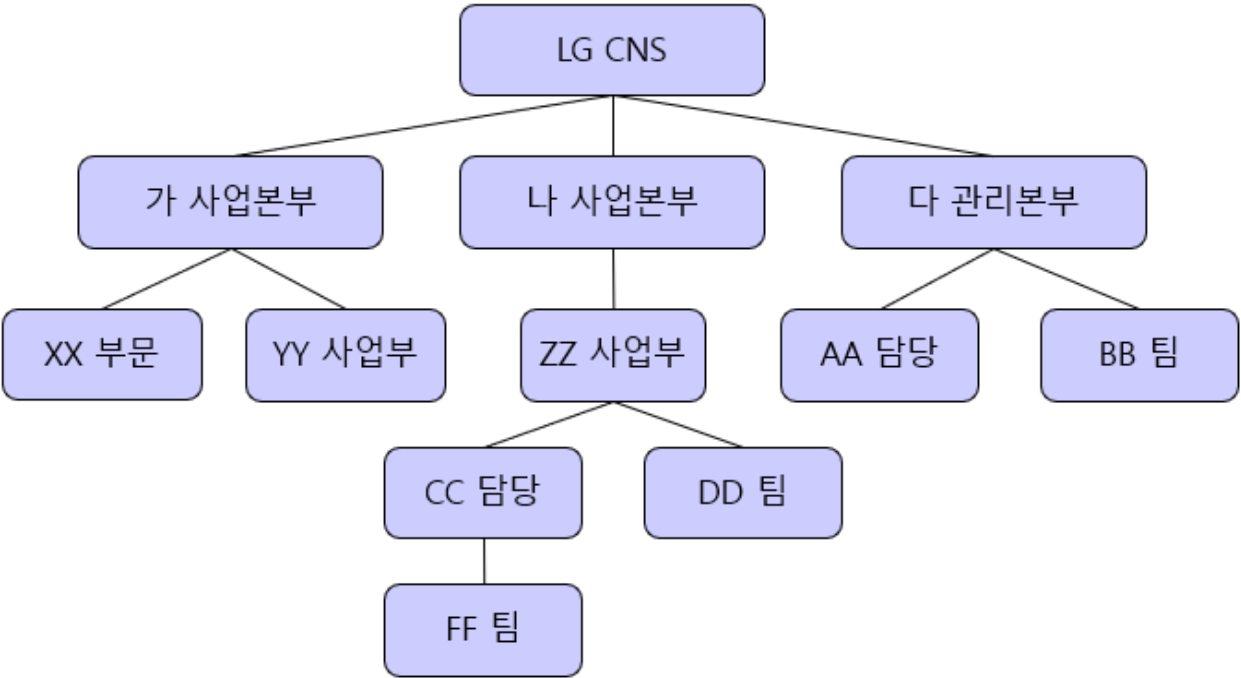
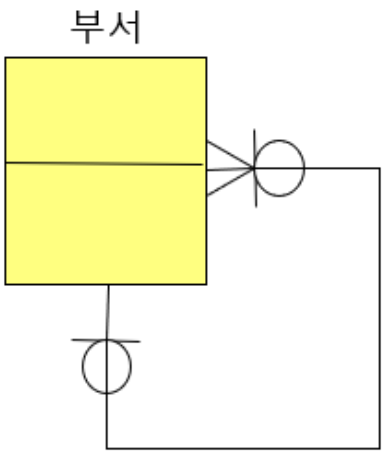
- 관계도출절차

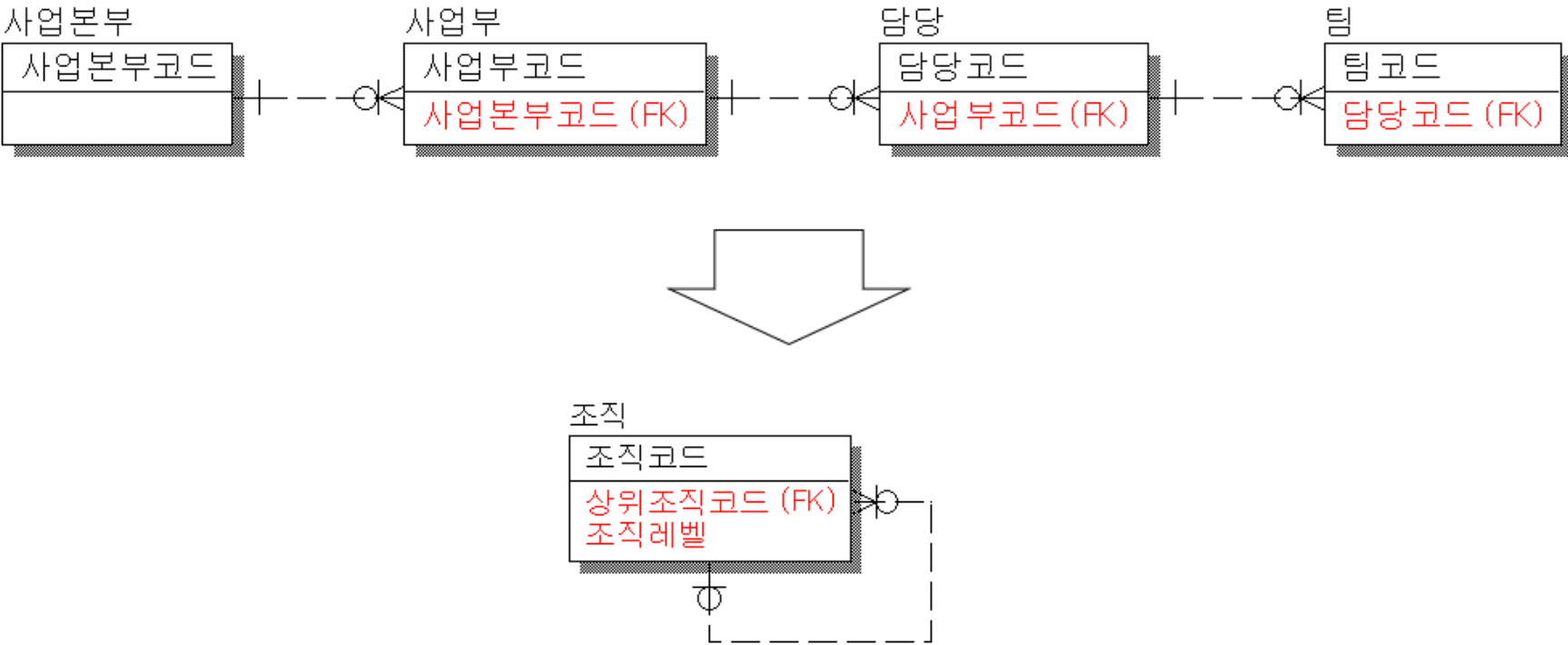
5. 고객과 질문하고 협의하여 모델을 검토하는 시간을 갖는다.



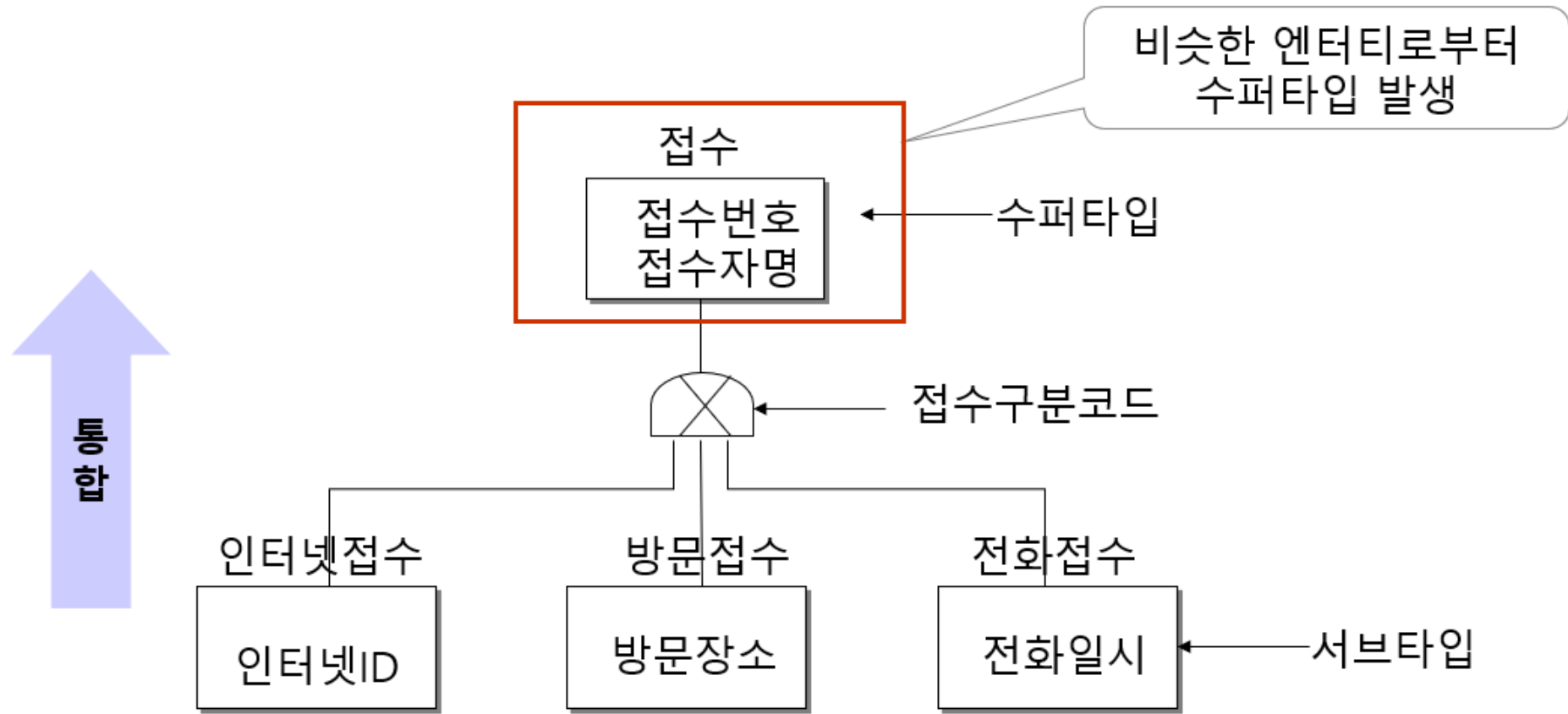
일반적 관계 Normal Relationship	M:M 관계	자기참조관계 Recursive Relationship	BOM관계 Bill of Material Relationship	병렬관계 Parallel Relationship
				

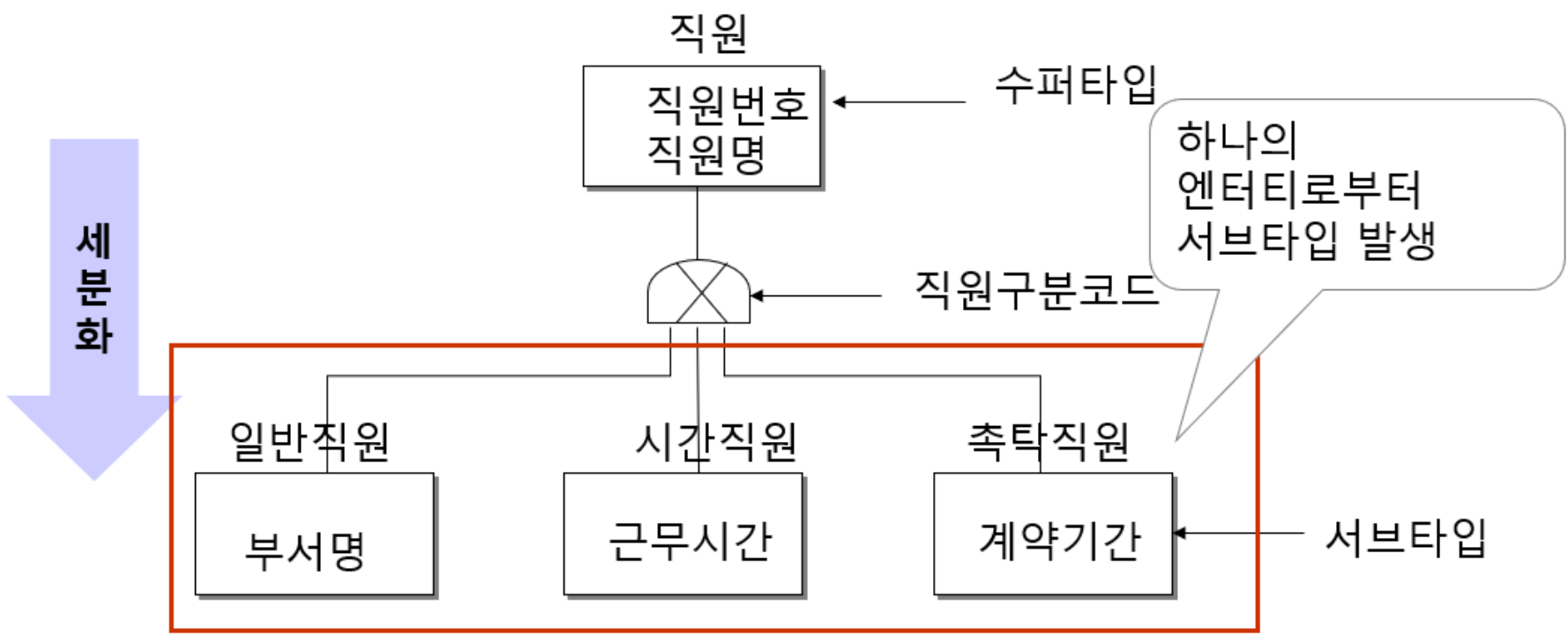
하나의 엔터티 내에서 어커런스와 어커런스가 1:M 관계를 맺고 있는 형태

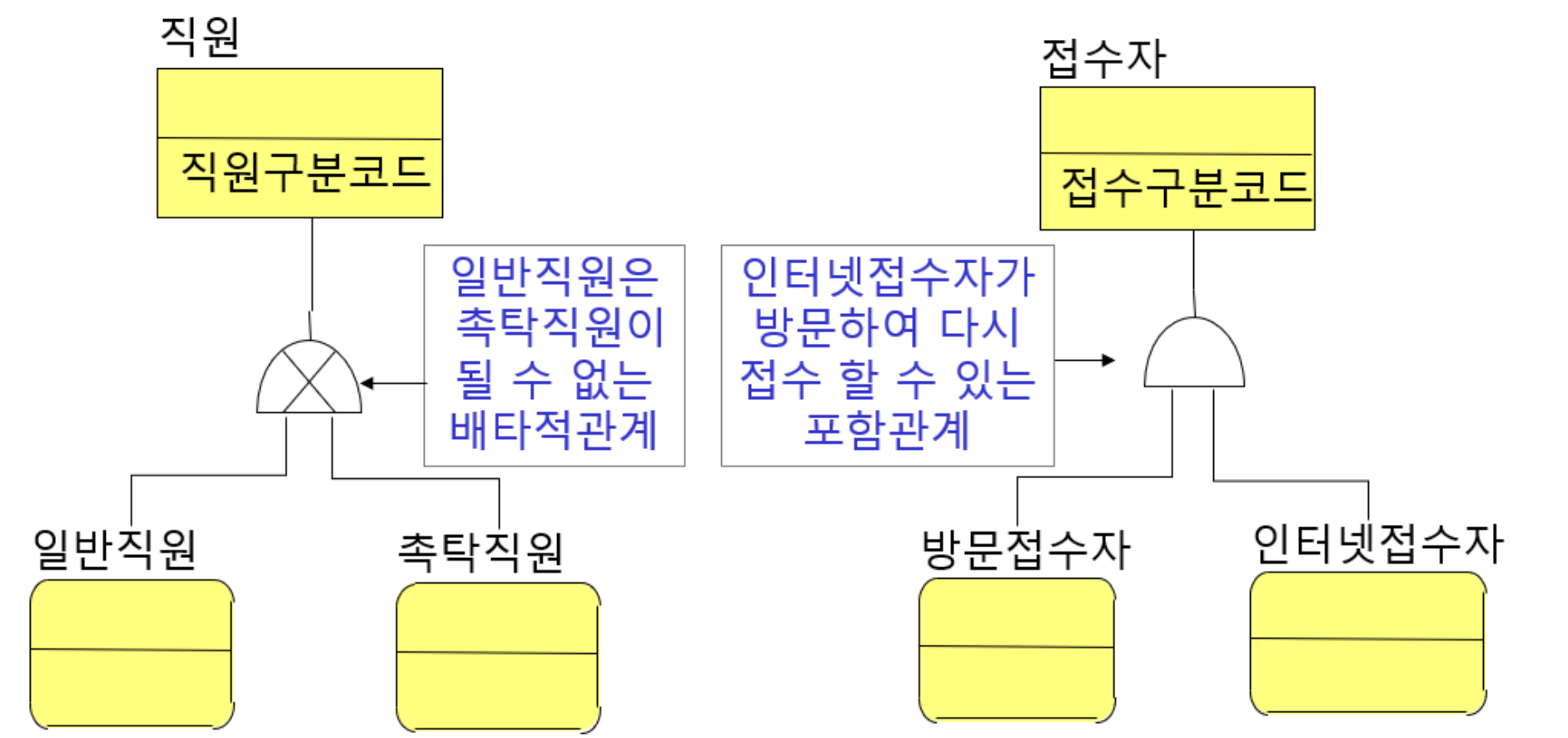




- ❖ 자기참조 관계는 일반적으로 조직, 메뉴, 게시물 관리에서 많이 사용함
- ❖ 레벨 속성을 두면 특정 레벨만 접근하기가 용이



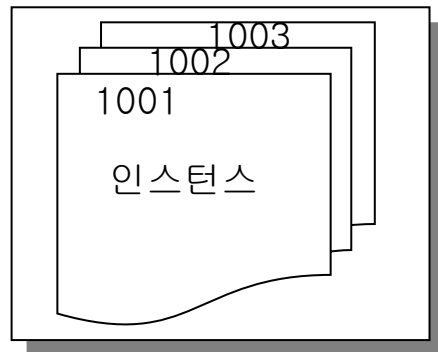




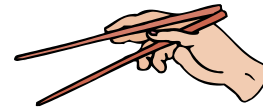
수퍼타입과 서브타입 사이에 1:1 관계형태로서 서브타입을 구분하는 방법에 따라 배타적관계와 포함관계로 구분됨

- 식별자(Identifiers) 개념

- ✓ 엔터티는 인스턴스들의 집합이라고 하였다. 여러 개의 집합체를 담고 있는 하나의 통에서 각각을 구분할 수 있는 논리적인 이름이 있어야 한다. 이 구분자를 식별자(Identifier)라고 한다.



주문 엔터티



식별자는 엔터티내에서 인스턴스들을 구분할 수 있는 구분자이다.

• 식별자의 특징

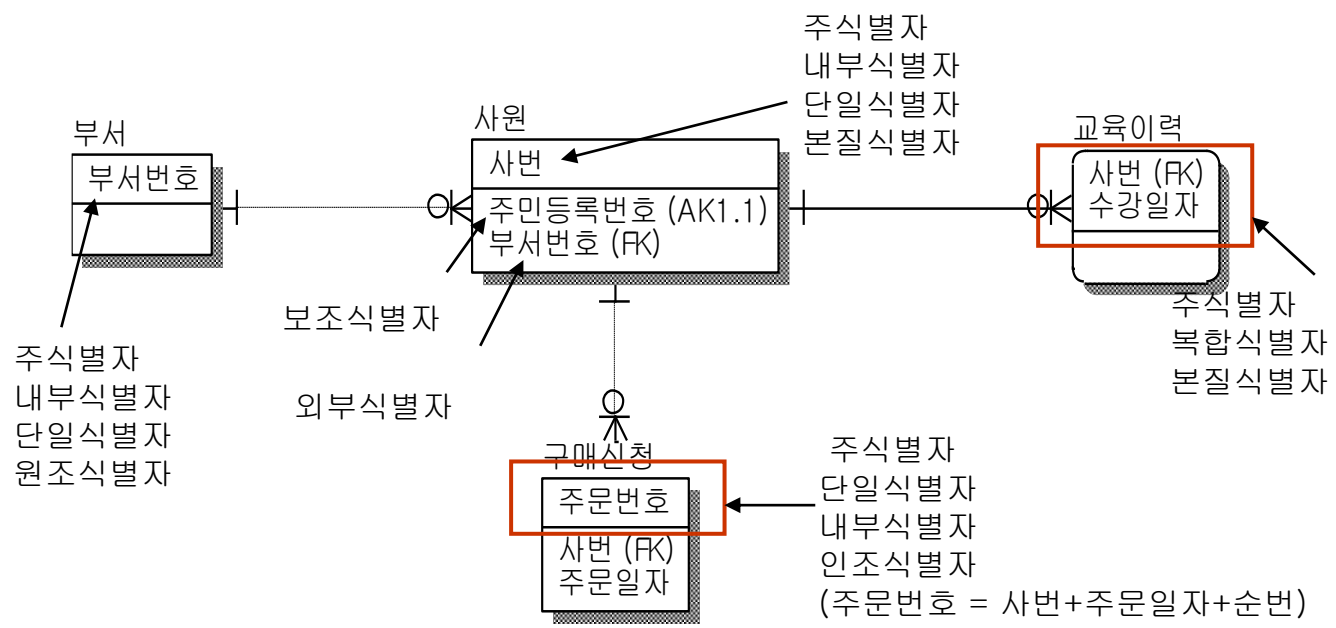
- ✓ 주식별자에 의해 엔터티내에 모든 인스턴스들이 유일하게 구분되어야 한다.
- ✓ 주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수가 되어야 한다.
- ✓ 지정된 주식별자의 값은 자주 변하지 않는 것이어야 한다.
- ✓ 주식별자가 지정이 되면 반드시 값이 들어와야 한다.

특징	내용	비고
유일성	주식별자에 의해 엔터티내에 모든 인스턴스들을 유일하게 구분함	예) 사원번호가 주식별자가 모든 직원들에 대해 개인별로 고유하게 부여됨
최소성	주식별자를 구성하는 속성의 수는 유일성을 만족하는 최소의 수가 되어야 함	예) 사원번호만으로도 고유한 구조인데 사원분류코드+사원번호로 식별자가 구성될 경우 부절한 주식별자 구조임
불변성	주식별자가 한 번 특정 엔터티에 지정되면 그 식별자의 값은 변하지 않아야 함	예) 사원번호의 값이 변한다는 의미는 이전기록이 말소되고 새로운 기록이 발생하는 개념임
존재성	주식별자가 지정되면 반드시 데이터 값이 존재 (Null 안됨)	예) 사원번호 없는 회사직원은 있을 수 없음

• 식별자 분류 및 표기법

분류	식별자	설명
대표성 여부	주식별자	엔터티 내에서 각 어커런스를 구분할 수 있는 구분자이며, 타 엔터티와 참조 관계를 연결할 수 있는 식별자
	보조식별자	엔터티 내에서 각 어커런스를 구분할 수 있는 구분자이나 대표성을 가지지 못해 참조관계 연결을 못함
스스로 생성여부	내부식별자	엔터티 내부에서 스스로 만들어지는 식별자
	외부식별자	타 엔터티와의 관계를 통해 타 엔터티로부터 받아오는 식별자
속성의 수	단일식별자	하나의 속성으로 구성된 식별자
	복합식별자	둘 이상의 속성으로 구성된 식별자
대체 여부	본질식별자	업무에 의해 만들어지는 식별자
	인조식별자	업무적으로 만들어지지는 않지만 원조식별자가 복잡한 구성을 가지고 있기 때문에 인위적으로 만든 식별자

• 식별자 표기법



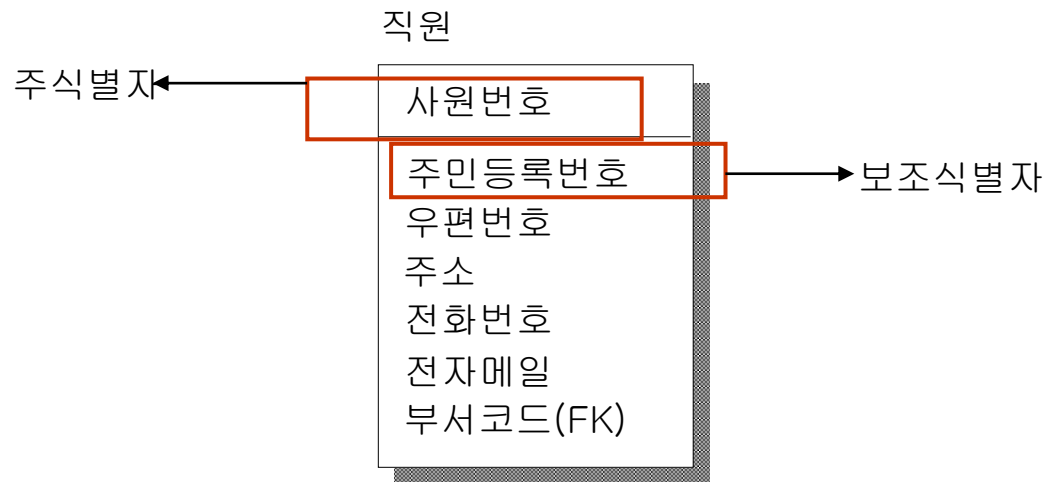
데이터모델에서 식별자(IDENTIFIER) 분류

- 주식별자 도출기준

- ✓ 해당 업무에서 자주 이용되는 속성을 주식별자로 지정한다.
- ✓ 명칭, 내역 등과 같이 이름으로 기술되는 것들은 가능하면 주식별자로 지정하지 않는다.
- ✓ 복합으로 주식별자로 구성할 경우 너무 많은 속성이 포함되지 않도록 한다.

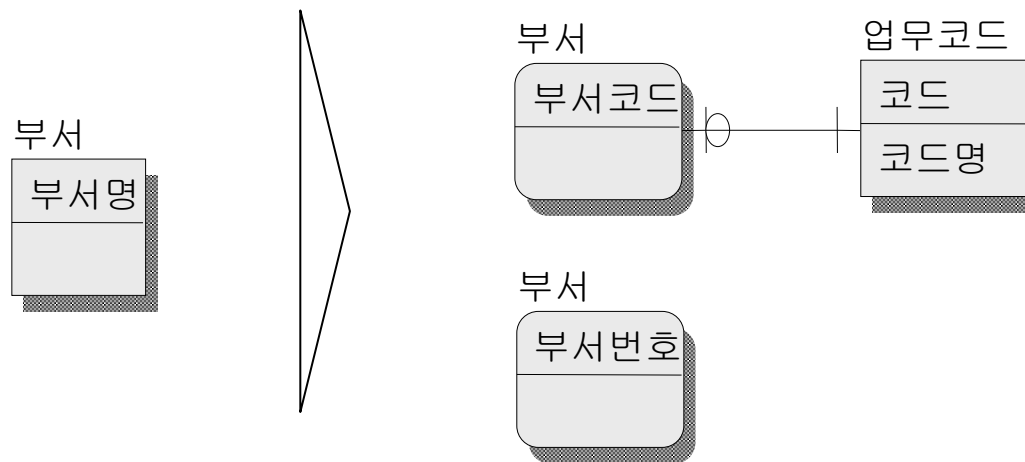
- 주식별자 도출기준 - 해당업무에서 자주이용

- ✓ 예를 들면, 직원이라는 엔터티가 있을 때 유일하게 식별가능한 속성으로는 주민등록번호와 사원번호가 존재할 수 있다. 사원번호가 그 회사에서 직원을 관리할 때 흔히 사용되므로 사원번호를 주식별자로 지정하고 주민등록번호는 보조식별자로 사용할 수 있다.



- 주식별자 도출기준 - 명칭이나 내역은 부적절

- ✓ 명칭, 내역 등과 같이 이름으로 기술되는 것들은 가능하면 주식별자로 지정하지 않도록 한다. 예를 들어, 한 회사에 부서이름이 100개가 있다고 할 때, 각각의 부서이름은 유일하게 구별될 수 있다고 하여 부서이름을 주식별자로 지정하지 않도록 해야 한다
- ✓ 명칭이나 내역이 있고 인스턴스들을 식별할 수 있는 다른 구분자가 존재하지 않을 경우는 새로운 식별자를 생성하도록 한다. 보통 일련번호와 코드를 많이 사용한다.



주식별자 도출기준 - 속성의 수가 많아지지 않도록 함

- ✓ 주식별자로 선정하기 위한 속성이 복합으로 구성되어 주식별자가 될 수 있을 때 가능하면 주식별자 선정하기 위한 속성의 수가 많지 않도록 해야 한다.
- ✓ 식별자의 속성의 개수가 많을 때(일반적으로 7~8개 이상)는 새로운 인조식별자(Artificial Identifier)를 생성하여 데이터 모델을 구성하는 것이 데이터 모델을 한층 더 단순하게 하고 애플리케이션을 개발할 때 조건절을 단순하게 할 수 있는 방법이 될 수 있다.

접수

접수일자
관할부서
입력자사번
접수방법코드
신청인구분코드
신청인주민번호
신청회수
신청자명
...

너무 많은 주식별자 속성

접수번호 =
관할부서(3)+접수일자(8)+
일련번호(3)

접수

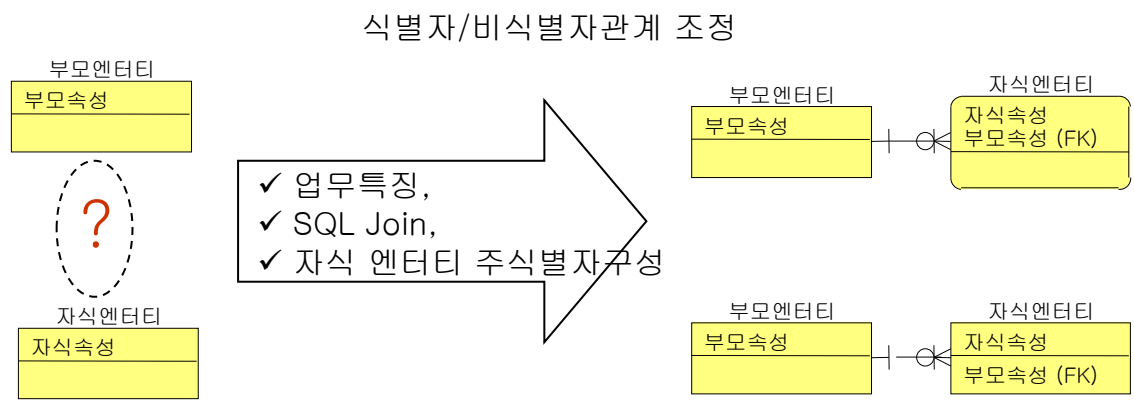
접수번호
신청자명
장소
계약금
접수일자
관할부서
입력자사번
접수방법코드
...

인조식별자를 통해 단순화한 주식별자 속성

```
SELECT 계약금
FROM 접수
WHERE 접수.접수일자 = '2010.07.15'
AND 접수.관할부서 = '1001'
AND 접수.입력자사번 = 'AB45588'
AND 접수.접수방법코드 = 'E'
AND 접수.신청인구분코드 = '01'
AND 접수.신청인주민번호 = '7007171234567'
AND 접수.신청횟수 = '1'
```

```
SELECT 계약금
FROM 접수
WHERE 접수.접수번호 = '100120100715001'
```


- 식별자관계와 비식별자관계에 따른 식별자
 - 외부식별자(Foreign Identifier)는 자기 자신의 엔터티에서 필요한 속성이 아니라 다른 엔터티와의 관계를 통해 자식 쪽에 엔터티에 생성되는 속성을 외부식별자라 하며 데이터베이스 생성 시에 Foreign Key역할을 한다.

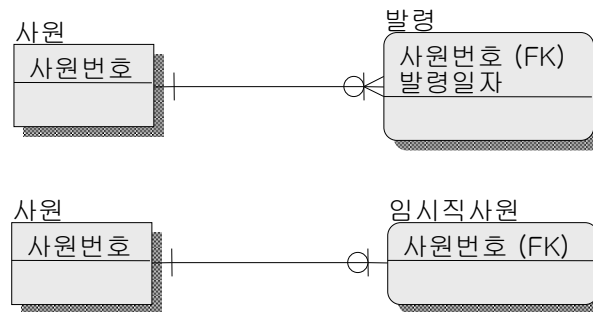


엔터티 사이 관계유형은 업무특징, 자식엔터티의 주식별자구성, SQL 전략에 의해 결정된다.

- 식별자관계

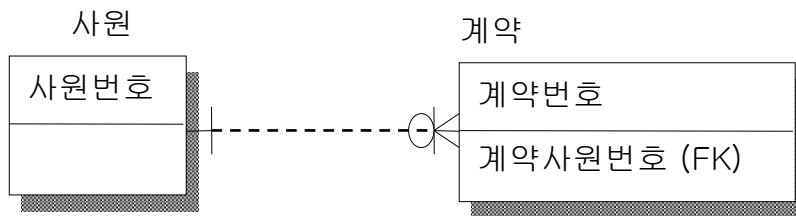
- ✓ 자식엔터티의 주식별자로 부모의 주식별자가 상속이 되는 경우를 식별자 관계(Identifying Relationship)라고 지칭한다.
- ✓ 부모로부터 받은 식별자를 자식엔터티의 주식별자로 이용하는 경우는 Null값이 오면 안되므로 반드시 부모엔터티가 생성되어야 자기 자신의 엔터티가 생성되는 경우이다.

외부식별자의 주식별자 역할



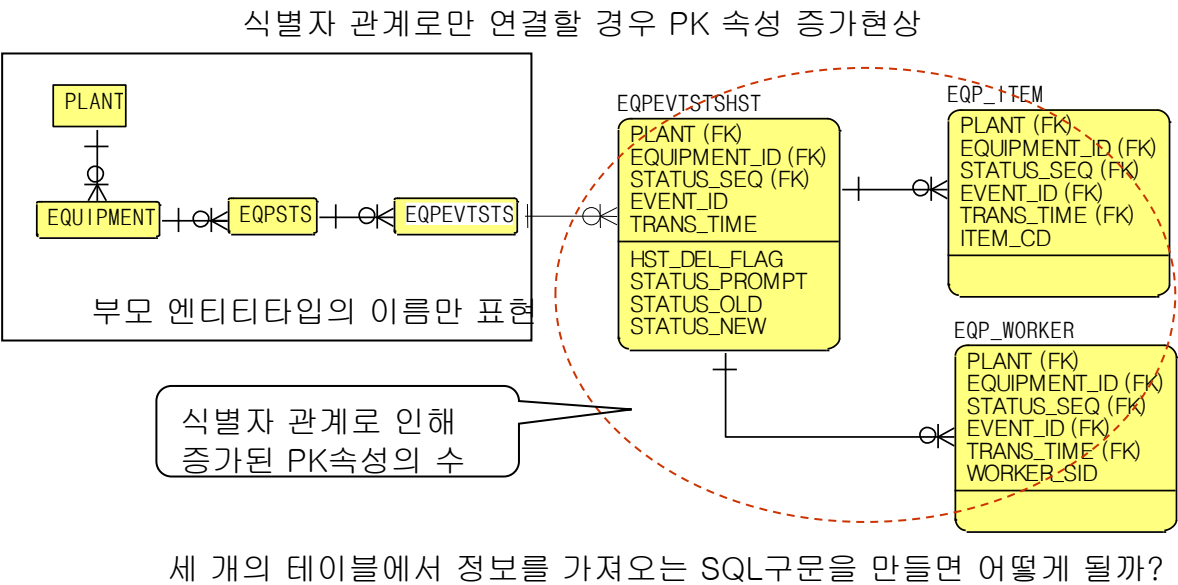
• 비식별자관계 (Non-Identifying Relationship) 의 정의

- ✓ 부모엔티티로부터 속성을 받았지만 자식엔티티의 주식별자로 사용하지 않고 일반적인 속성으로만 사용하는 경우가 있다. 이와 같은 경우를 비식별자 관계(Non-Identifying Relationship)라고 하며 다음의 네 가지 경우에 비식별자 관계에 의한 외부속성을 생성한다.
- ✓ 자식엔티티에서 받은 속성이 반드시 필수가 아니어도 무방하기 때문에 부모 없는 자식이 생성될 수 있는 경우이다.
- ✓ 엔티티별로 데이터의 생명주기(Life Cycle)를 다르게 관리할 경우이다. 예를 들어 부모엔티티에 인스턴스가 자식의 엔티티와 관계를 가지고 있었지만 자식만 남겨두고 먼저 소멸될 수 있는 경우가 이에 해당된다. 이에 대한 방안으로 물리데이터베이스 생성 시 Foreign Key를 연결하지 않는 임시적인 방법을 사용하기도 하지만 데이터 모델상에서 관계를 비식별자관계로 조정하는 것이 가장 좋은 방법이다.
- ✓ 여러 개의 엔티티가 하나의 엔티티로 통합되어 표현되었는데 각각의 엔티티가 별도의 관계를 가질 때이며 이에 해당된다.
- ✓ 자식엔티티에 주식별자로 사용하여도 되지만 자식엔티티에서 별도의 주식별자를 생성하는 것이 더 유리하다고 판단될 때 비식별자 관계에 의한 외부식별자로 표현한다.



식별자 관계로만 설정할 경우의 문제점

- 식별자 관계를 연결한 데이터 모델의 PK속성의 수는 데이터 모델의 흐름이 길어질수록 증가할 수 밖에 없는 구조를 가지게 된다.



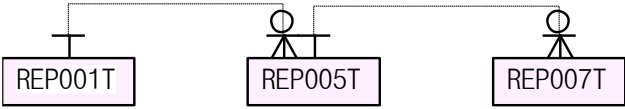
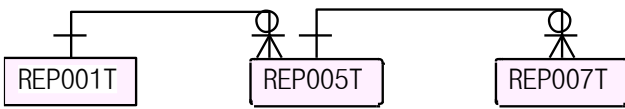
원 부모엔티티 : 1개
2대 부모엔티티 : 2개 이상 = 원부모 1개 + 추가 1개 이상 +
3대 부모엔티티 : 3개 이상 = 원부모 1개 + 2대 1개 + 추가 1개 이상
3대 부모엔티티 : 3개 이상 = 원부모 1개 + 2대 1개 + 3대 1개 + 추가 1개 이상
4대 부모엔티티 : 4개 이상 = 원부모 1개 + 2대 1개 + 3대 1개 + 4 1개 + 추가 1개 이상

- 식별자 관계로만 설정할 경우의 문제점

- ✓ 고작 3개 정도의 엔터티를 조인했을 뿐인데 SQL구문의 WHERE절이 매우 길어진 사실을 확인할 수 있다. 실제로 프로젝트에서는 개발자가 개발할 때 당연히 데이터 모델을 참조하면서 엔터티와 관계를 이용하여 개발해야 하는데 생성된 엔터티 스키마 정보만을 보고 개발하는 경우가 많다.

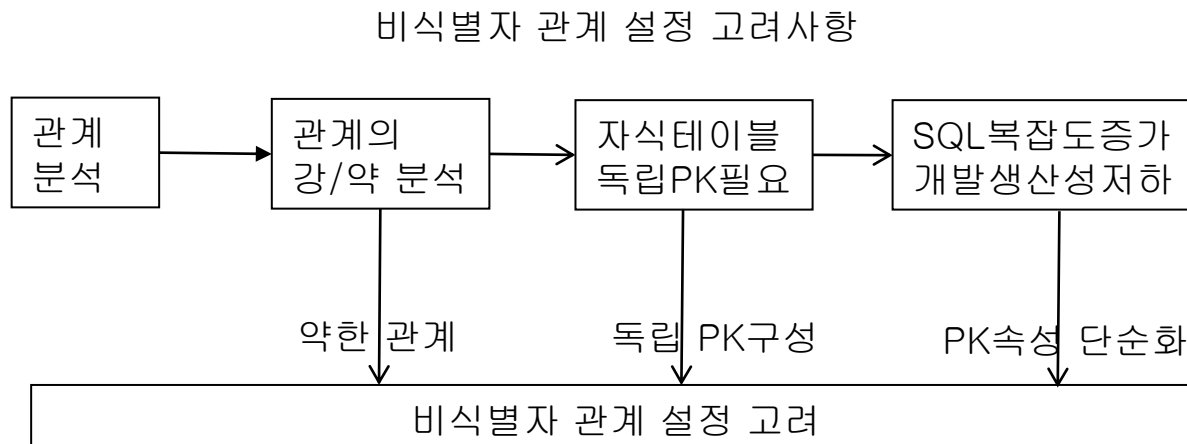
```
SELECT  A.EVENT_ID, A.TRANS_TIME, A.HST_DEL_FLAG, A.STATUS_PROMPT,  
        A.STATUS_OLD, A.STATUS_NEW  
FROM EQPEVTSTSHST A, EQP_ITEM B, EQP_WORKER C  
WHERE A.PLANT= B.PLANT  
      AND A.EQUIPMENT_ID= B.EQUIPMENT_ID  
      AND A.STATUS_SEQ= B.STATUS_SEQ  
      AND A.EVENT_ID= B.EVENT_ID  
      AND A.TRANS_TIME= B.TRANS_TIME  
      AND B.ITEM_CD= 'A001'  
      AND A.PLANT = C.PLANT  
      AND A.EQUIPMENT_ID= C.EQUIPMENT_ID  
      AND A.STATUS_SEQ= C.STATUS_SEQ  
      AND A.EVENT_ID= C.EVENT_ID  
      AND A.TRANS_TIME= C.TRANS_TIME  
      AND C.WORKER_SID= 'A012008001'
```

- 식별자와 비식별자관계에 따른 SQL 비교

비식별자관계	식별자 관계
	
<pre>SELECT A.기본가중치, A.조정가중치 FROM REP007T A, REP005T B, REP001T C WHERE A.분야번호 = B.분야번호 AND B.점검번호 = C.점검번호 AND C.점검번호 = '105'</pre>	<pre>SELECT A.기본가중치, A.조정가중치 FROM REP007T A WHERE A.점검번호 = '105'</pre>
테이블 REP007T에서 점검번호='301'인 데이터를 조회	

• 식별자관계와 비식별자관계 모델링

- ✓ 실제로 프로젝트를 전개할 때 식별자관계와 비식별자관계를 취사선택하여 연결하는 내공은 높은 수준의 데이터모델링의 기술이라 할 수 있다.
- ✓ 식별자관계에서 비식별자관계를 파악하는 기술이 필요한데 다음 흐름(Flow)에 따라 비식별자관계를 선정한다면 합리적으로 관계를 설정하는 모습이 될 수 있다. 기본적으로 식별자관계로 모든 관계가 연결되면서 다음 조건에 해당할 경우 비식별자관계로 조정하면 된다.



• 식별자와 비식별자관계 비교

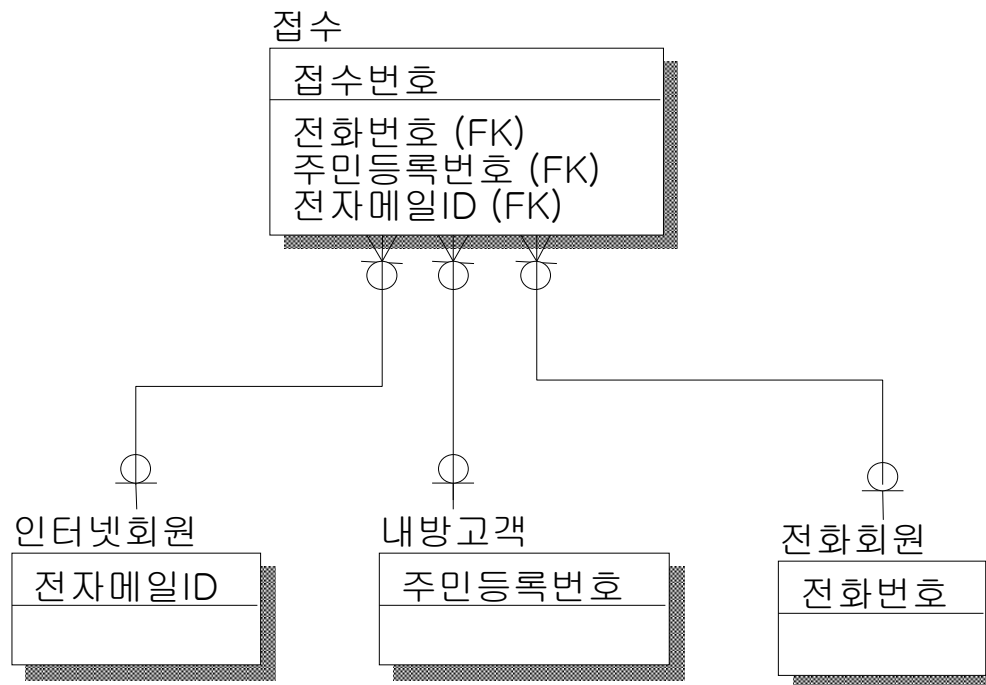
항목	식별자관계	비식별자관계
목적	강한 연결관계 표현	약한 연결관계 표현
자식 주식별자 영향	자식 주식별자의 구성에 포함됨	자식 일반 속성에 포함됨
표기법	실선 표현	점선 표현
연결 고려사항	<ul style="list-style-type: none">- 반드시 부모엔터티 종속- 자식 주식별자구성에 부모 주식별자포함 필요- 상속받은 주식별자속성을 타 엔터티에 이전 필요	<ul style="list-style-type: none">- 약한 종속관계- 자식 주식별자구성을 독립적으로 구성- 자식 주식별자구성에 부모 주식별자 부분 필요- 상속받은 주식별자속성을 타 엔터티에 차단 필요- 부모쪽의 관계참여가 선택관계

- 비식별자 관계를 사용하는 경우

- ① 여러 개의 엔터티가 하나의 엔터티로 통합되어 표현될 경우
(각각은 식별자 관계로 하여도 무방)
- ② 자식 엔터티에 별도의 주식별자를 가지고 있는 것이 더 유리하다고 판단되는 경우
- ③ 자식과 관련이 있는 엔터티로의 주식별자 상속을 차단하기 위해서
- ④ 부모 없는 자식이 생성될 수 있는 경우

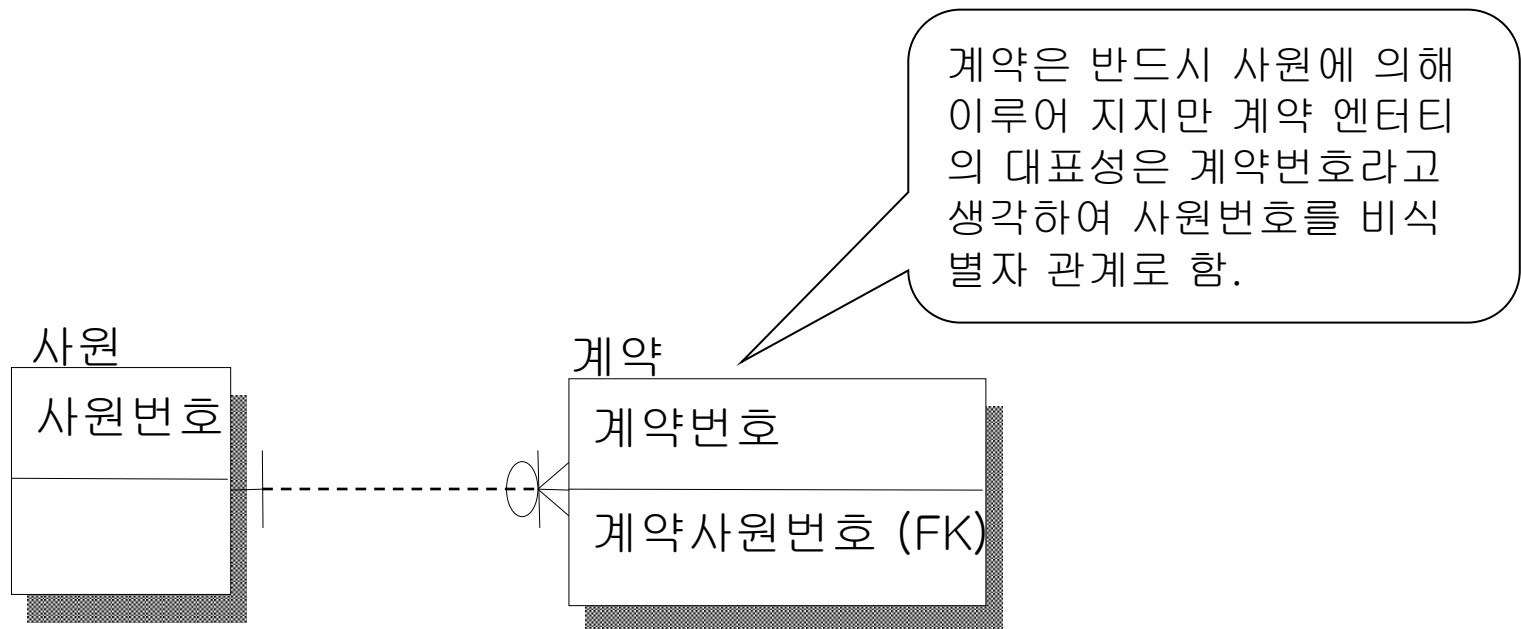
■ 비식별자 관계를 사용하는 경우

① 여러 개의 엔터티가 하나의 엔터티로 통합되어 표현될 경우



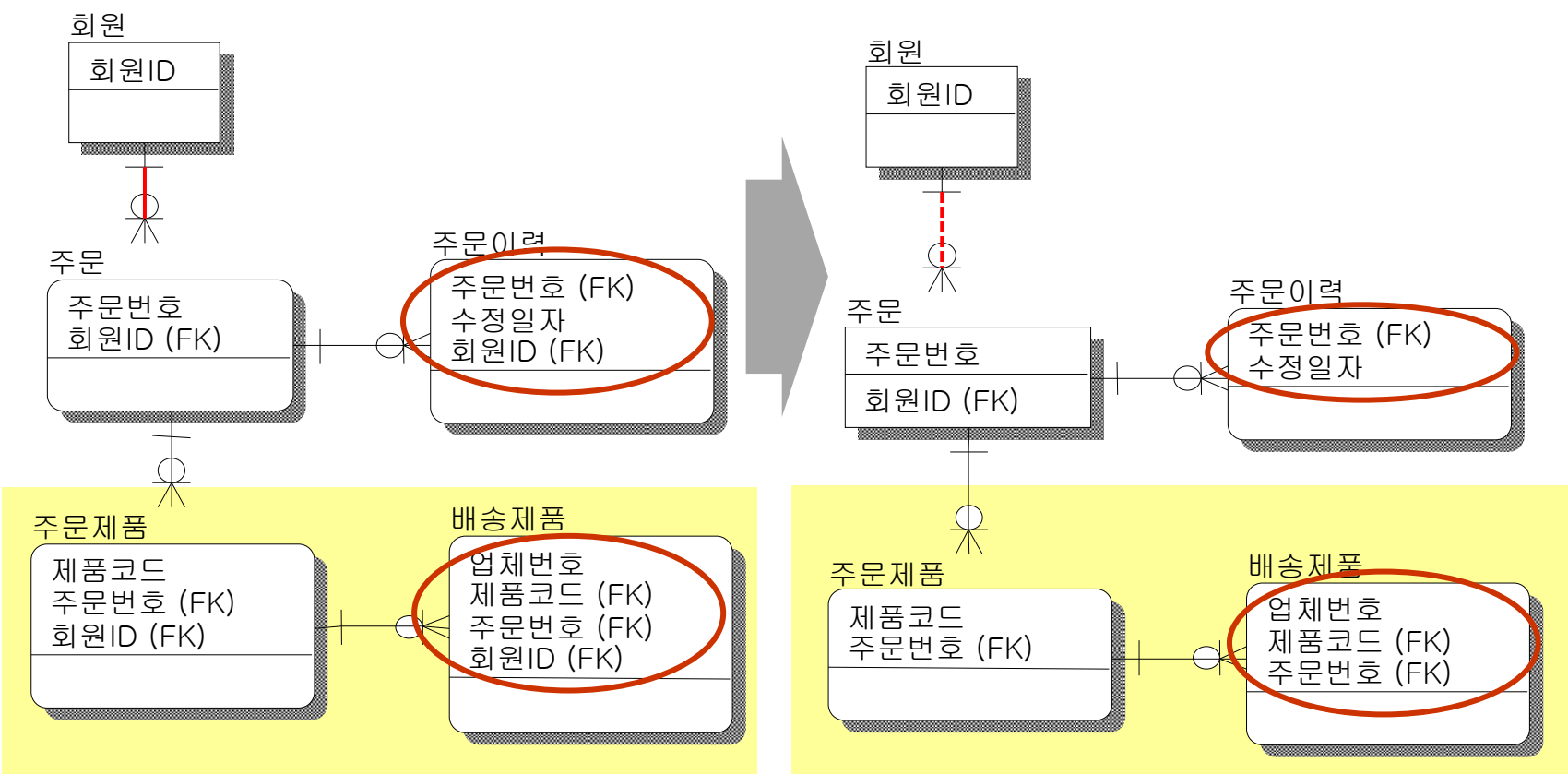
- 비식별자 관계를 사용하는 경우

- ② 자식 엔터티에 별도의 주식별자를 가지고 있는 것이 더 유리하다고 판단되는 경우



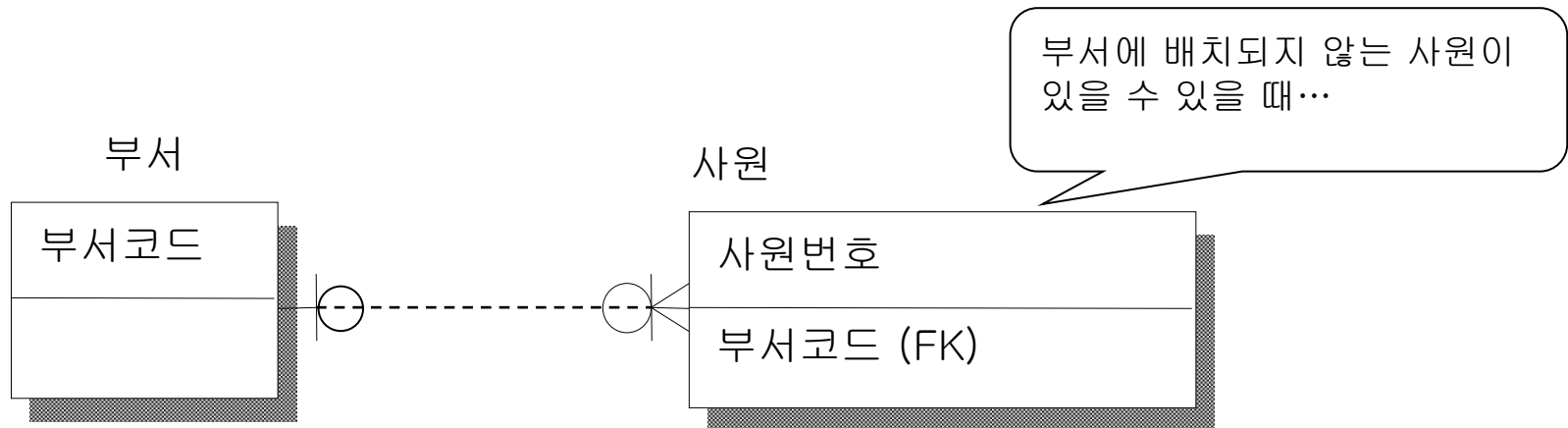
비식별자 관계를 사용하는 경우

③ 자식과 관련이 있는 엔터티로의 주식별자 상속을 차단하기 위해서



- 비식별자 관계를 사용하는 경우

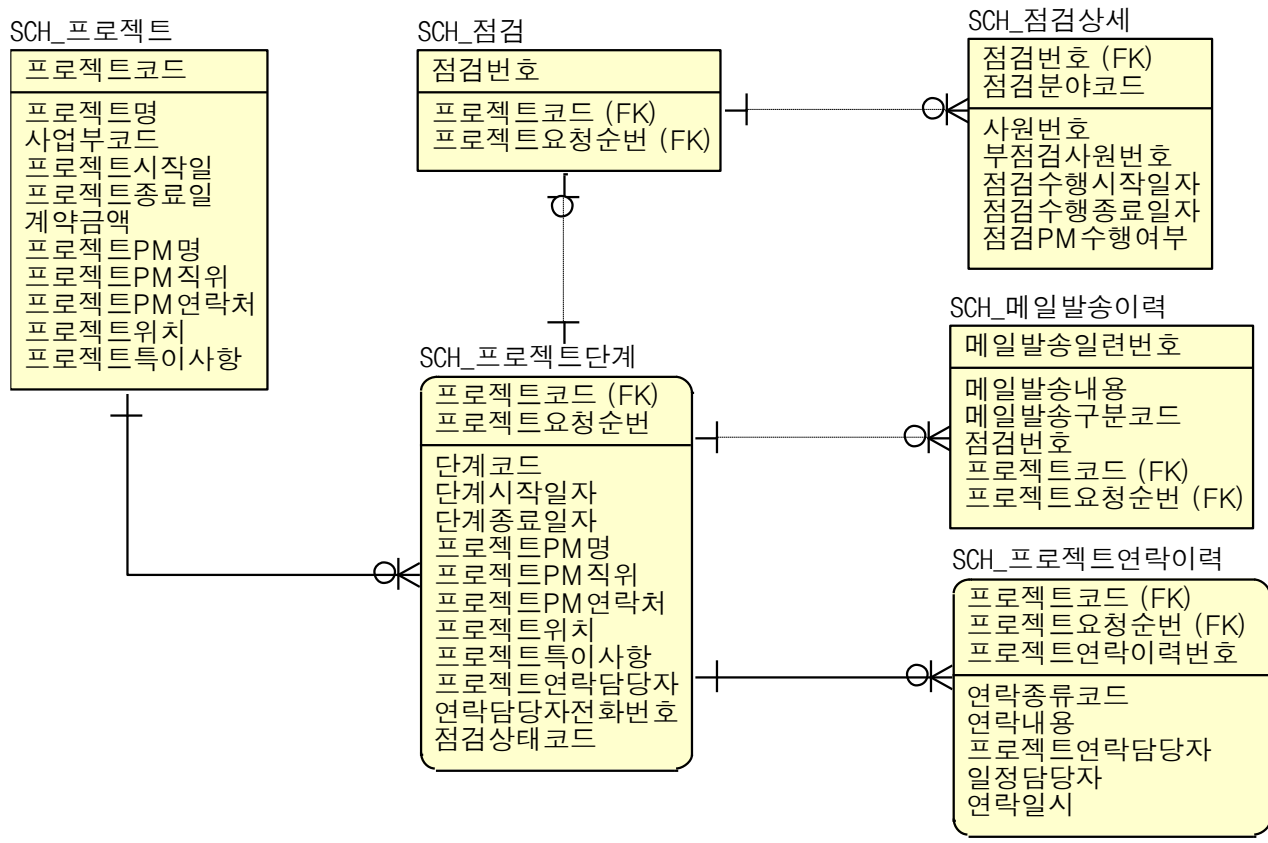
- ④ 부모 없는 자식이 생성될 수 있는 경우



* 본 사례는 예외적인 것이므로 꼭 필요한 경우에만 사용

• 식별자와 비식별자를 적용한 데이터 모델

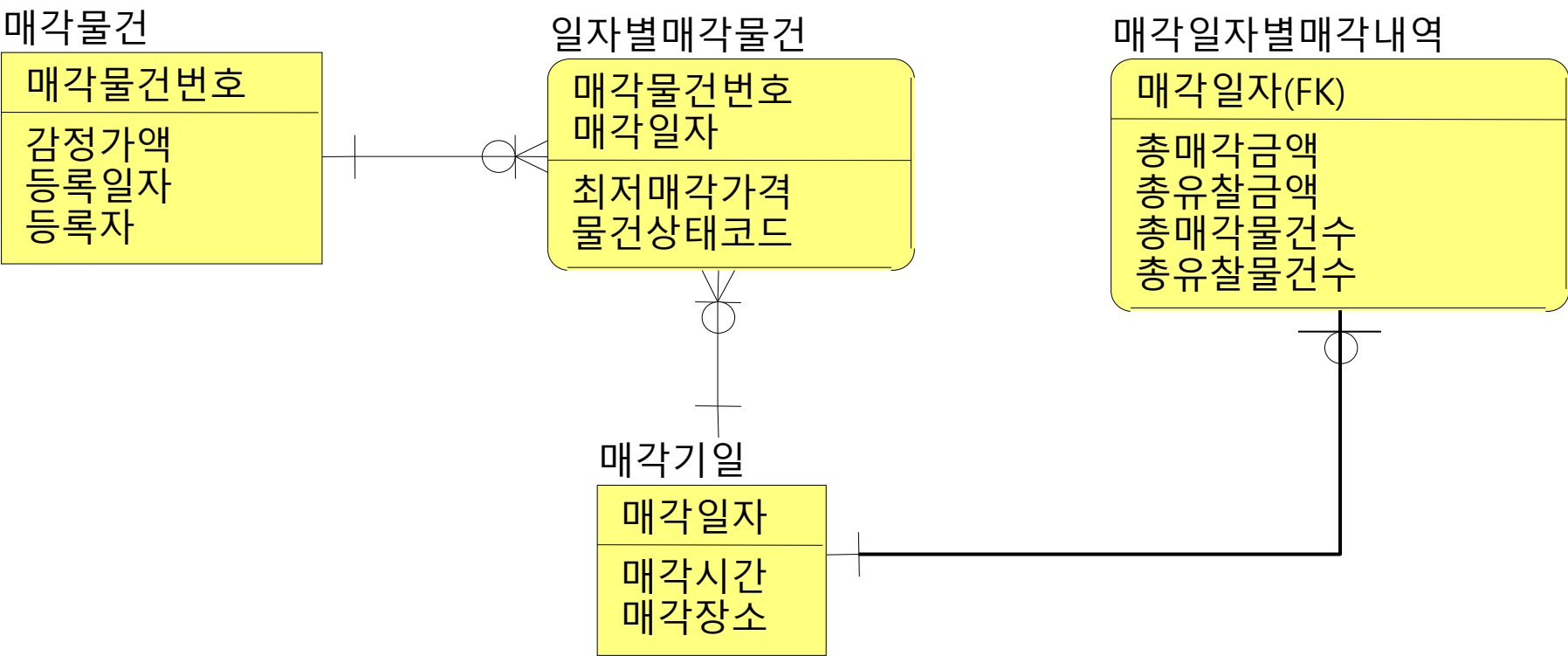
✓ 식별자와 비식별자 관계가 적절하게 설정된 데이터 모델은 아래와 같이 균형감 있게 나타난다.



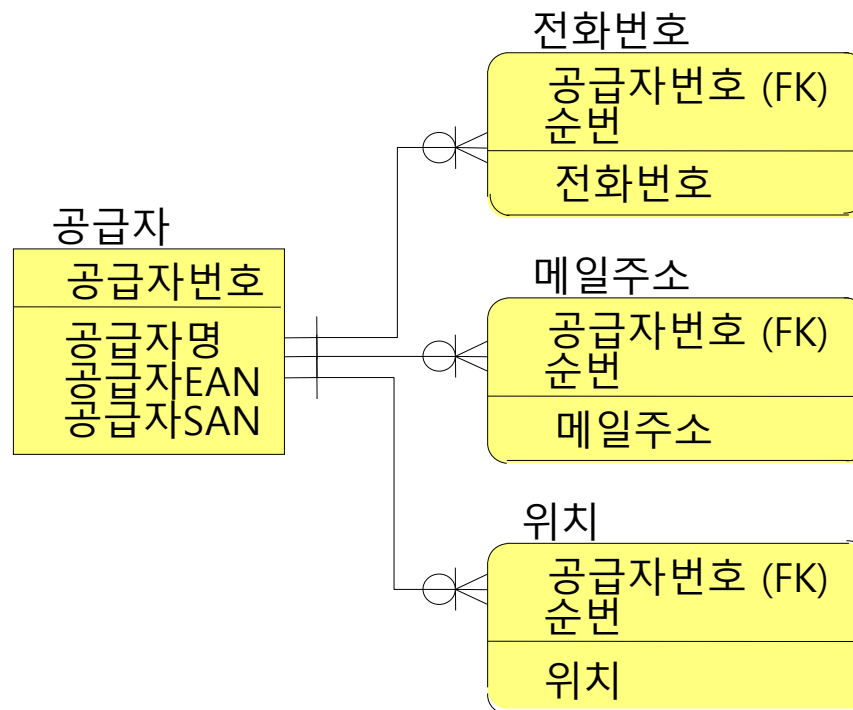


1. 다양한 유형의
2. 속성
3. 관계
4. 식별자

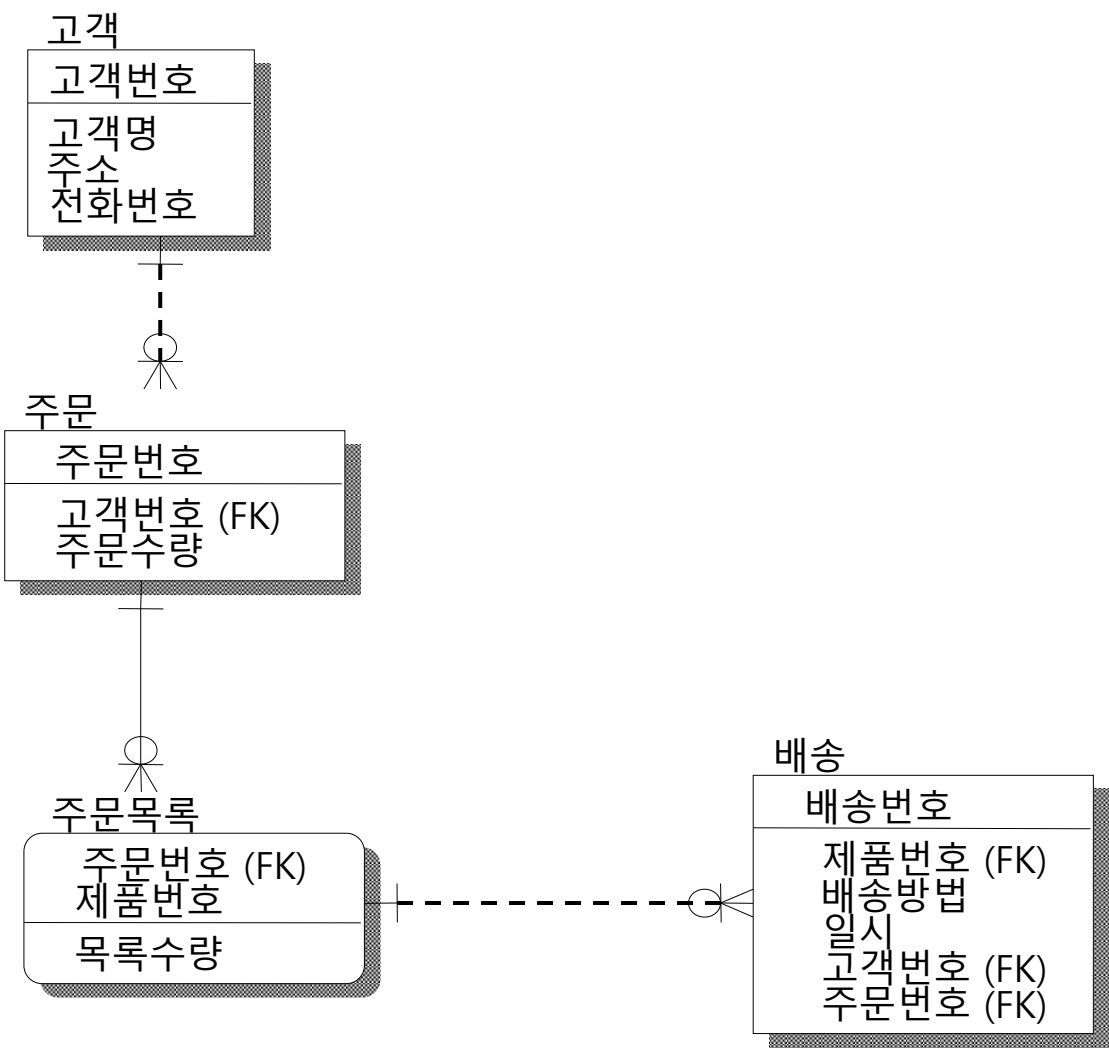
- 다음 데이터모델을 해석해 보자



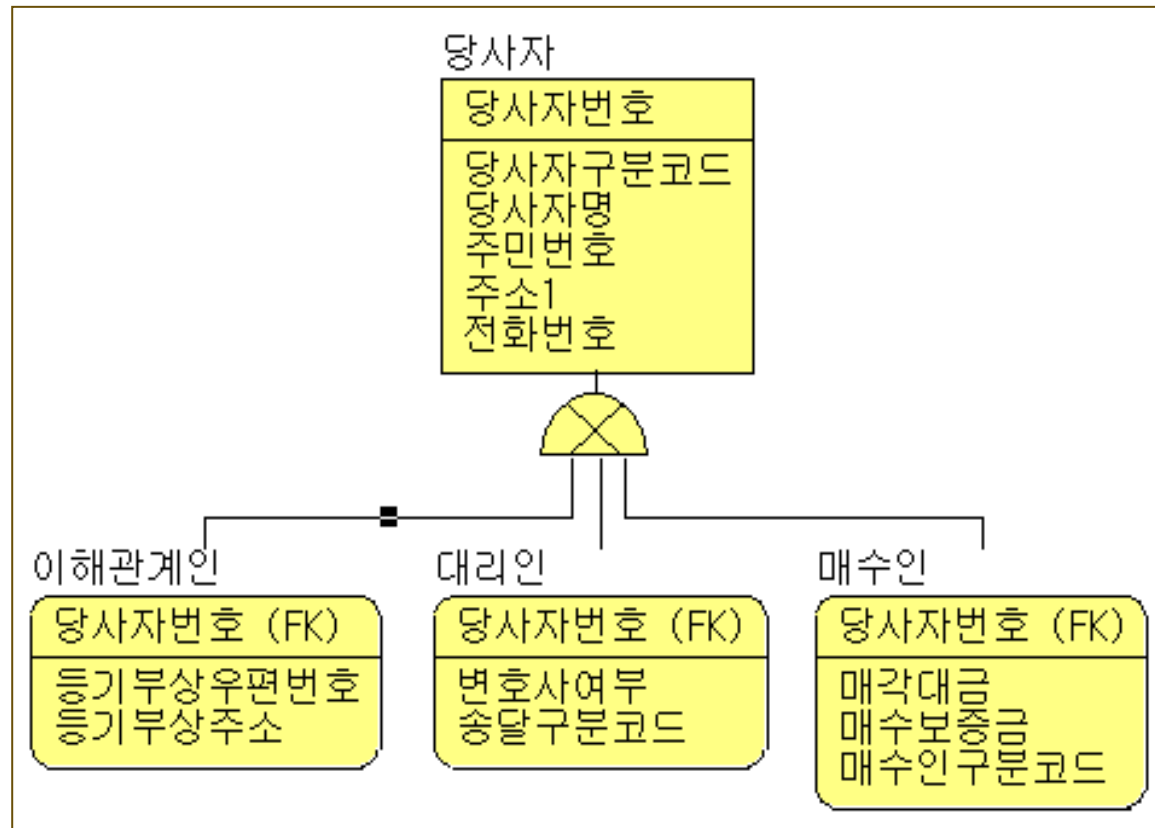
- 다음 데이터모델을 해석해 보자



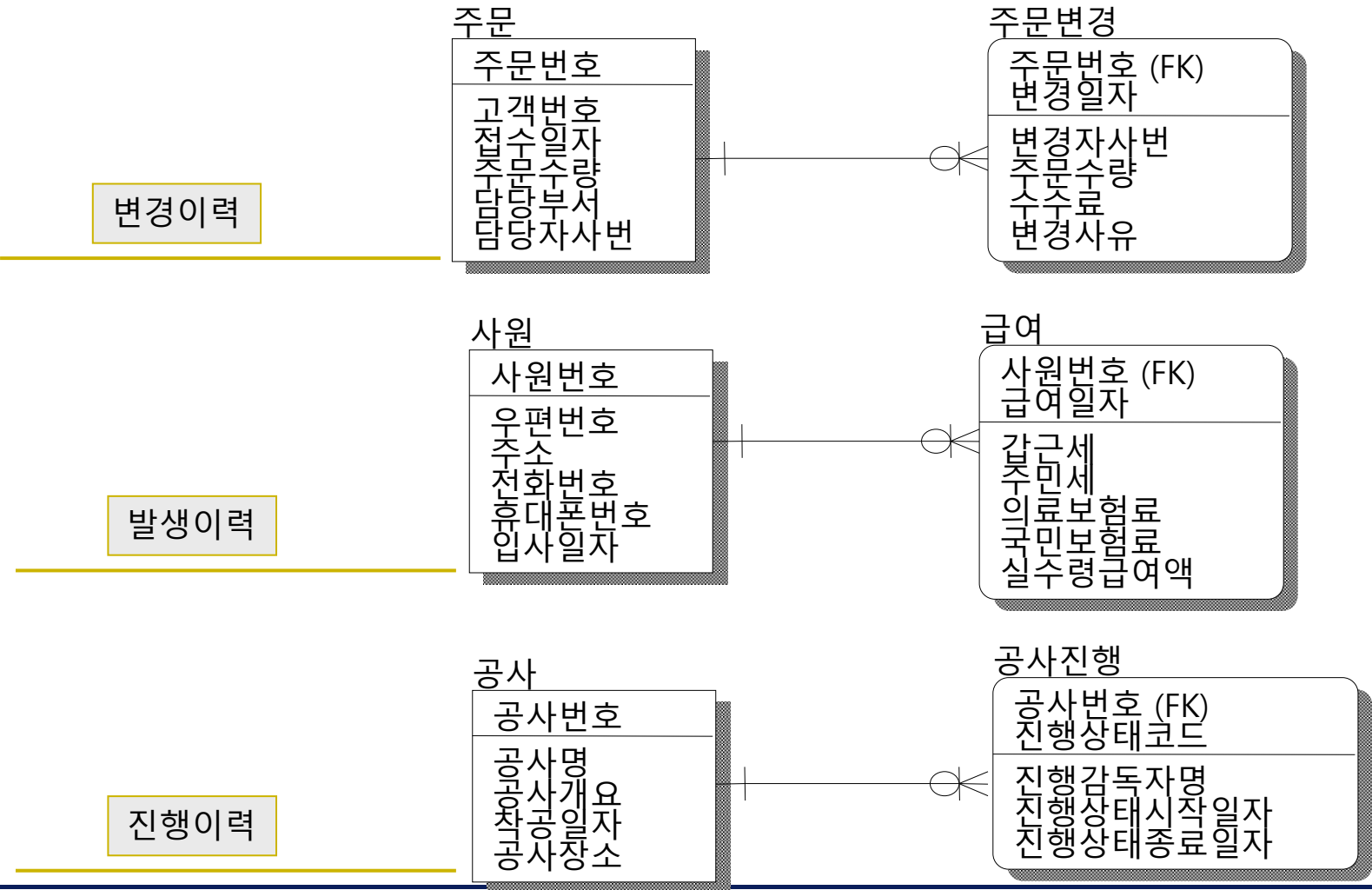
- 다음 데이터모델을 해석해 보자



- 다음 데이터모델을 해석해 보자

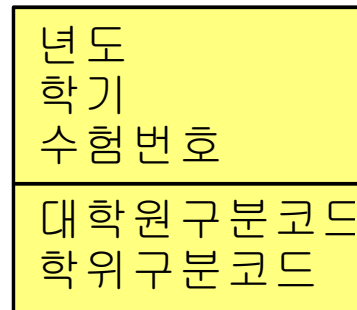


- 다음 데이터모델을 해석해 보자

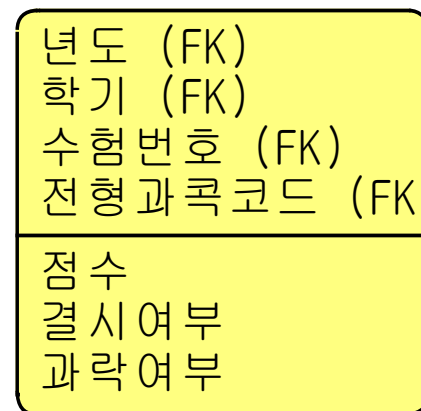


- 다음 데이터모델을 해석해 보자

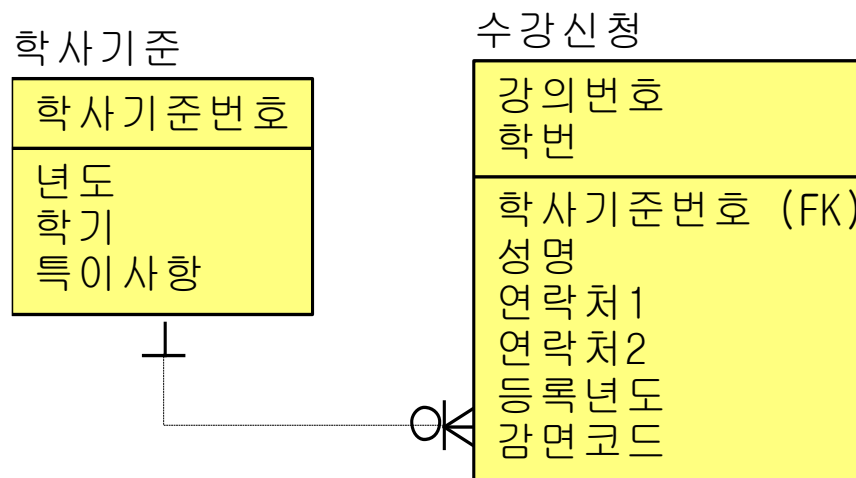
입시마스터



전형과목실적



- 다음 데이터모델을 해석해 보자



```
SELECT COUNT(B.학번)
FROM 학사기준 A, 수강신청 B
WHERE A.학사기준번호 = B.학사기준번호
AND A.년도 = '2008'
AND A.학기 = '3'
```

