

DB프로그래밍

12주차

담당교수: 김희숙
(jasmin11@hanmail.net)

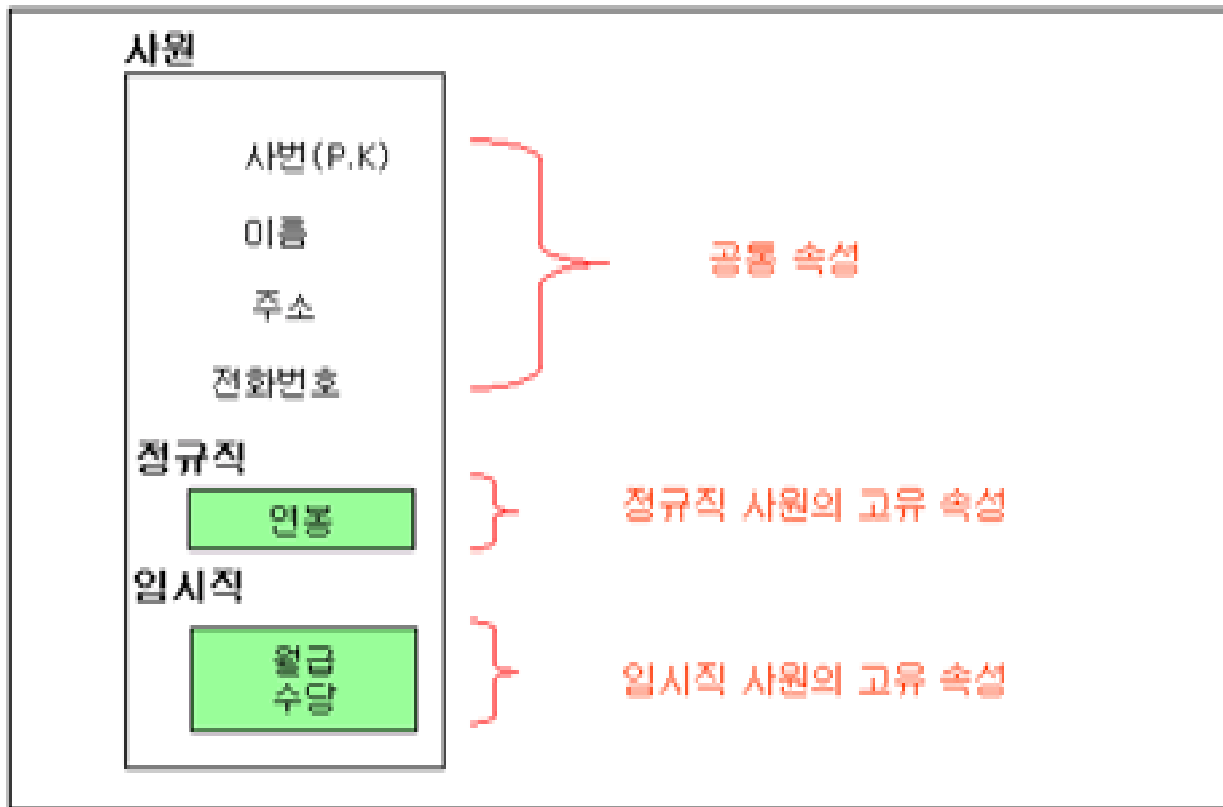


수퍼타입/서브타입

담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 수퍼타입/서브타입

- ❖ 상위 엔티티로 공통속성을 가진 것을 생성하고,
- ❖ 하위 엔티티에서 상위 엔티티의 식별자를 상속받고 하위 엔티티의 별도 속성을 갖게 되는 구조



- ❖ 수퍼타입 엔티티
상호 배타적인 더 작은 그룹으로 분할되는 엔티티
- ❖ 서브타입 엔티티
수퍼타입 내의 분해된 그룹

[요약] 수퍼타입

❖ 수퍼타입 단위

(서브 타입 각각이 갖고 있는 고유 속성의 개수가 적을 때 유리)

사원구분: 속성의 값이 0 이면 정규직, 1 이면 임시직

테이블 명: 사원

컬럼 명	사번	이름	주소	전화번호	사원구분	연봉	월급	수당
키 형태	PK							
견본 데이터	1	홍길동	서울	02)111-1111	0	2,000	NULL	NULL
	2	박찬호	대구	043)222-2222	0	1,500	NULL	NULL
	3	박세리	부산	031)-333-3333	1	NULL	80	5
	4	선동열	광주	02)444-4444	0	2,200	NULL	NULL

견본 데이터

--전체 사원 조회

```
SELECT * FROM 사원;
```

--정규직 사원 조회

```
SELECT 사번, 이름, 주소, 전화번호, 연봉  
FROM 사원  
WHERE 사원구분 = 0;
```

--임시직 사원 조회

```
SELECT 사번, 이름, 주소,전화번호, 월급, 수  
당  
FROM 사원  
WHERE 사원구분 = 1;
```

[요약] 서브타입

❖ 서브 타입 단위

검색시 사원을 임시직/정규직 사원으로 구분하는 경우가 많거나,

서브 타입 간에 업무적으로 독립적인 업무 활용 패턴을 갖을 경우에 유리

❖ 서브 타입 각각이 관리하는 고유 속성의 분포차가 심할 때 유리)

테이블 명 : 정규직 사원

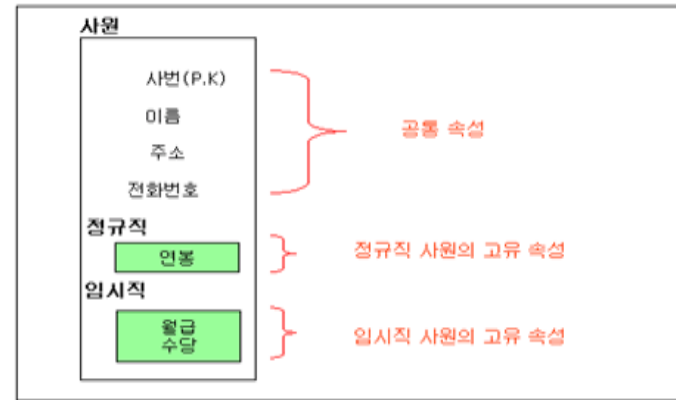
컬럼 명	사번	이름	주소	전화번호	연봉
키 형태	PK				
견본 데이터	1	홍길동	서울	02) 111-1111	2,000
	2	박찬호	대구	043) 222-2222	1,500
	3	선동렬	광주	02) 444-4444	2,200

테이블 명 : 임시직 사원

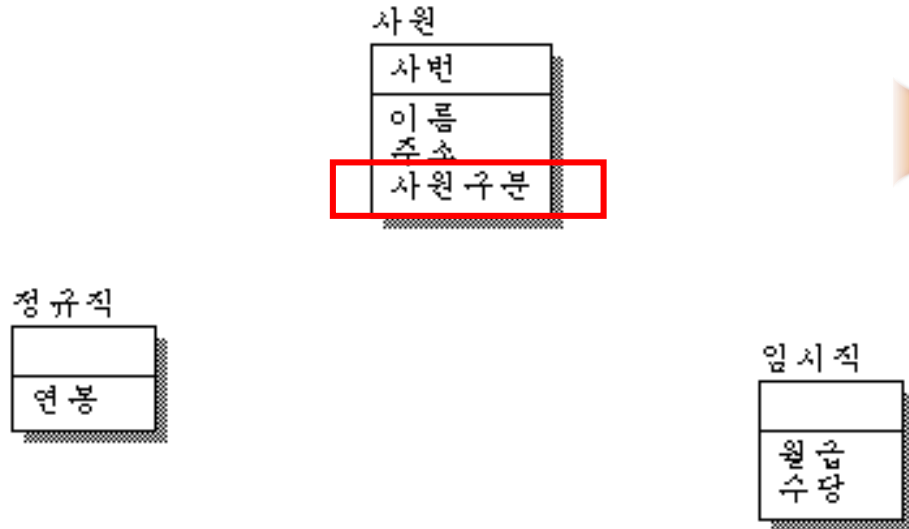
컬럼 명	사번	이름	주소	전화번호	월급	수당
키 형태	PK					
견본 데이터	1	박세리	부산	031)-333-3333	80	5

```
SELECT 사번, 이름, 주소 FROM 정규직사원  
UNION ALL  
SELECT 사번, 이름, 주소 FROM 임시직사원;
```

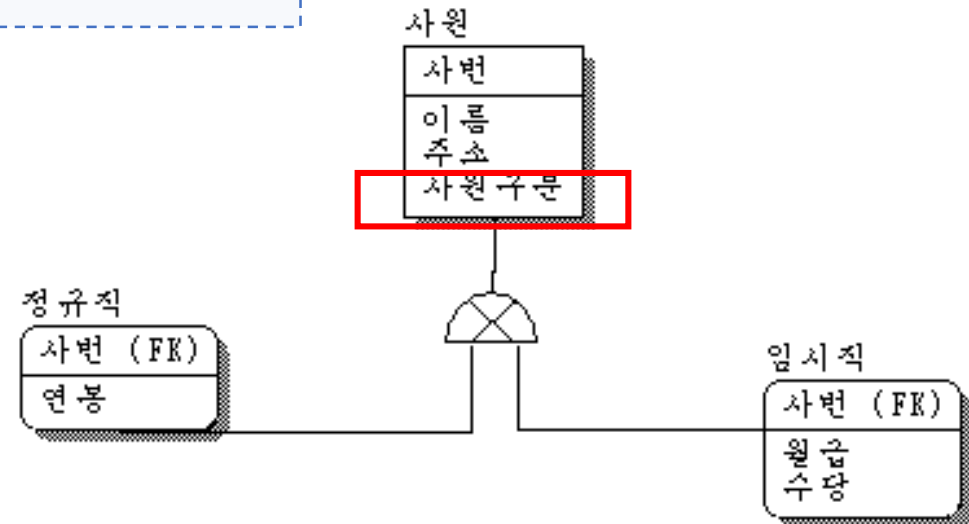
[예제] (사원) 수퍼타입/서브타입



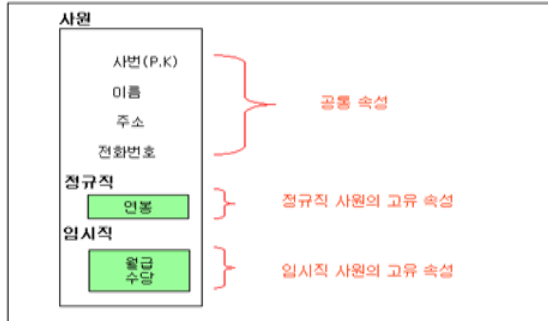
수퍼타입 작성 이전



수퍼타입 작성 이후



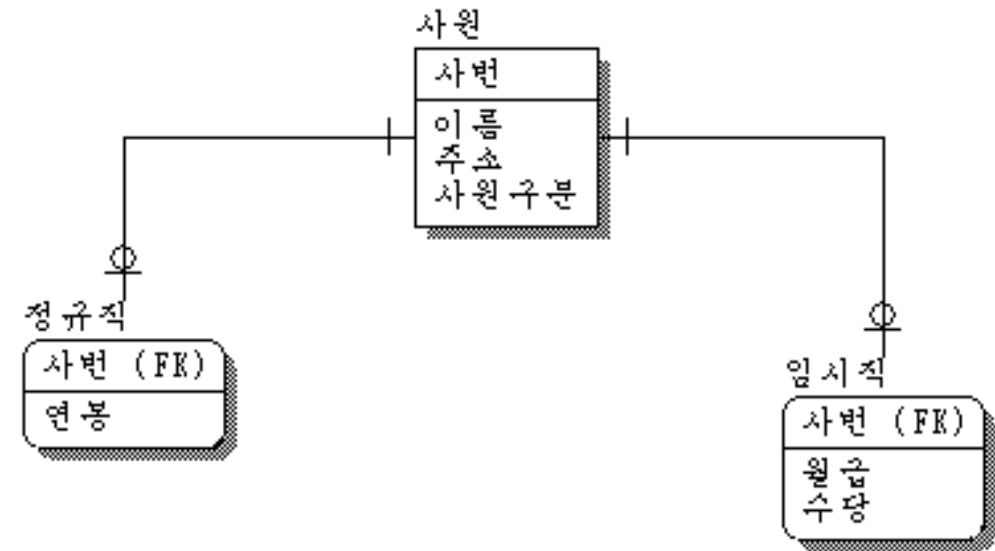
[예제] (사원) 수퍼타입/서브타입



❖ Supertype-Subtype Identity: 3개 개체

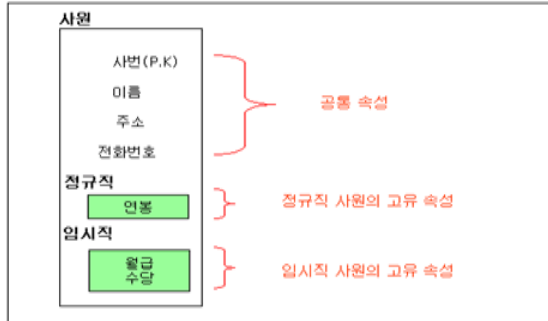
* 수퍼타입/서브타입을 테이블로 생성하는 3가지 방법

1. 테이블 3개 생성하는 경우,
사원(사번, 이름, 주소)
정규직(사번, 연봉)
임시직(사번, 월급, 수당)
2. 테이블 2개 생성하는 경우,
정규직(사번, 이름, 주소, 연봉)
임시직(사번, 이름, 주소, 월급, 수당)
3. 테이블 1개 생성하는 경우,
사원(사번, 이름, 주소, 사원구분, 연봉, 월급, 수당)



3개의 엔티티로 나뉘어 저장
프로그램 관리 복잡해질 수 있다

[예제] (사원) 수퍼타입/서브타입



❖ Supertype-Subtype Rolldown: 2개 개체

* 수퍼타입/서브타입을 테이블로 생성하는 3가지 방법

1. 테이블 3개 생성하는 경우,
사원(사번, 이름, 주소)
정규직(사번, 연봉)
임시직(사번, 월급, 수당)
2. 테이블 2개 생성하는 경우,
정규직(사번, 이름, 주소, 연봉)
임시직(사번, 이름, 주소, 월급, 수당)
3. 테이블 1개 생성하는 경우,
사원(사번, 이름, 주소, 사원구분, 연봉, 월급, 수당)

정규직

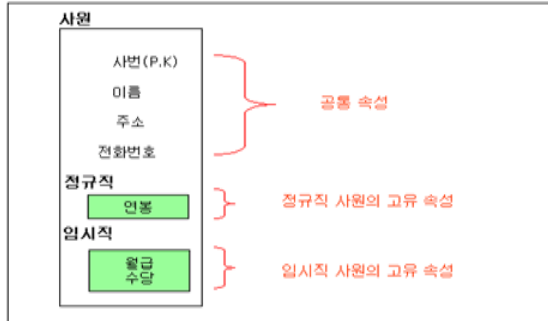
사번
연봉
이름
주소
사원구분

임시직

사번
월급
수당
이름
주소
사원구분

각각의 서브타입으로 분할

[예제] (사원) 수퍼타입/서브타입



❖ Supertype-Subtype Rollup: 1개 개체 통합

* 수퍼타입/서브타입을 테이블로 생성하는 3가지 방법

1. 테이블 3개 생성하는 경우,
사원(사번, 이름, 주소)
정규직(사번, 연봉)
임시직(사번, 월급, 수당)
2. 테이블 2개 생성하는 경우,
정규직(사번, 이름, 주소, 연봉)
임시직(사번, 이름, 주소, 월급, 수당)
3. 테이블 1개 생성하는 경우,
사원(사번, 이름, 주소, 사원구분, 연봉, 월급, 수당)

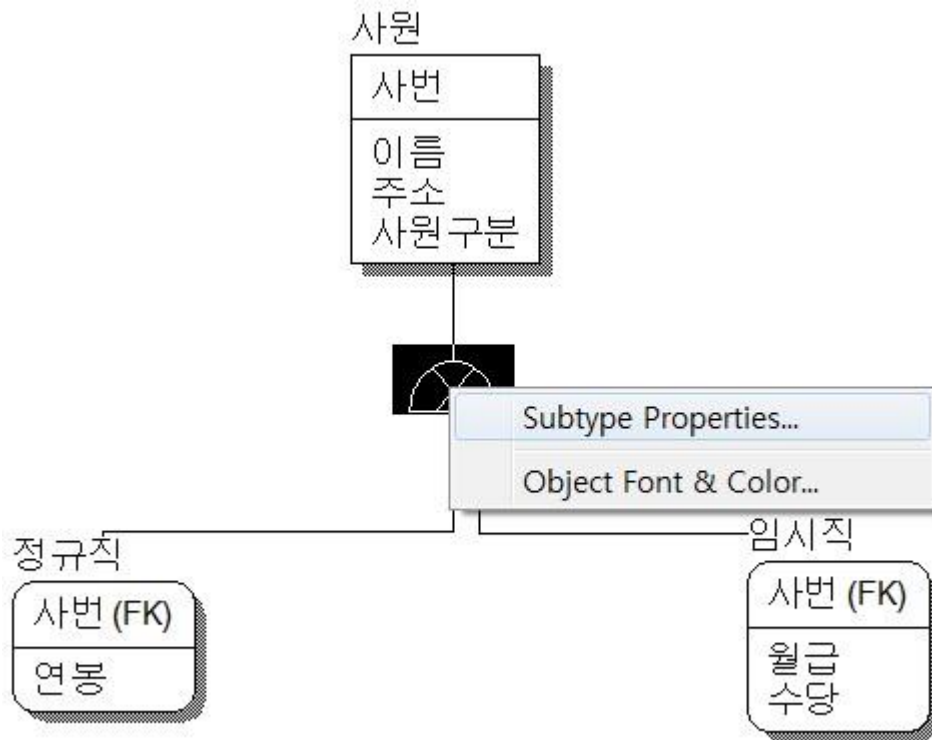
사원

사번
이름
주소
사원구분
월급
수당
연봉

구분코드 속성 필요: 사원구분

[요약] 수퍼타입/서브타입 구분

- ❖ Exclusive : 서로 배타적 관계
- ❖ Inclusive : 서로 혼용할 수 있는 포함 관계



Supertype: 사원

Subtypes: 정규직, 임시직

Type: ☒ Exclusive ☐ Inclusive

Discriminator:

- 사원
 - 사번
 - 이름
 - 주소
 - 사원구분

[예제] (자동차매매) 슈퍼타입/서브타입

[슈퍼/서브] 다음을 Erwin 논리적 설계 작성하시오

회원은 일반구매 회원과 딜러구매 회원으로 나뉜다.

일반구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받는다.

딜러구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받고 추가로 매물 목록, 신용등급, 소속회사 연락처를 받는다.

차량은 차대번호, 제조사, 모델, 등급, 연식, 변속기(수동, 자동), 도색, 주행거리, 가격 정보를 갖는다.

차량은 차종에 따라 승용차(소형, 중형, 대형), 화물차(트레일러, 윙바디 등), 승합차(6인승, 9인승, 12인승 등), 스포츠카(300마력, 400마력, 500마력 등)으로 분류할 수 있다.

모든 거래는 저장되며 거래 내용은 구입일시, 남은 보증일수, 게시기간, 회원등급 정보가 기록된다.

업체는 매물차량을 등록하기 전에 반드시 진단점검을 받아야 한다. 여기서 진단점검은 차량의 상태를 나타내기 위한 최소한의 검사이다.

검사항목으로는 차종확인, 사고여부, 주행진단, 계기판수리/교체흔적, 등록증확인, 차량내부진단, 차량외부진단, 옵션진단, 엔진룸진단, 하부진단이다.

[예제] (자동차매매) 슈퍼타입/서브타입

회원은 일반구매 회원과 딜러구매 회원으로 나뉜다.

일반구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받는다.

딜러구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받고 추가로 매물목록, 신용등급, 소속회사 연락처를 받는다.

차량은 차대번호, 제조사, 모델, 등급, 연식, 변속기(수동, 자동), 도색, 주행거리, 가격 정보를 갖는다.

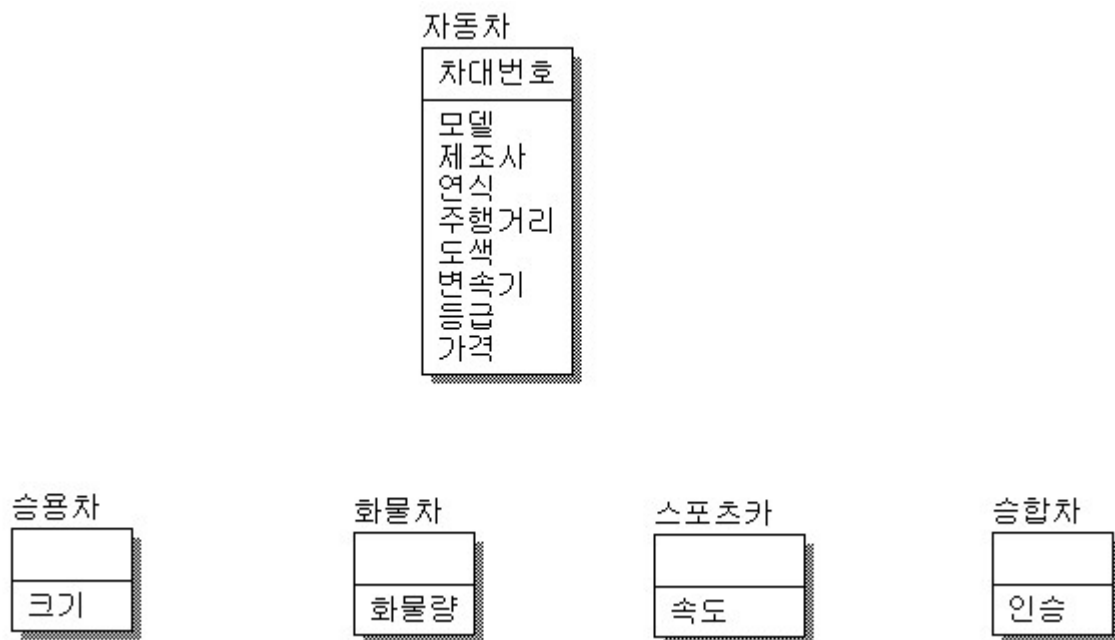
차량은 차종에 따라 승용차(소형, 중형, 대형), 화물차(트레일러, 윙바디 등), 승합차(6인승, 9인승, 12인승 등), 스포츠카(300마력, 400마력, 500마력 등)으로 분류할 수 있다.

모든 거래는 저장되며 거래 내용은 구입일시, 남은 보증일수, 게시기간, 회원등급 정보가 기록된다.

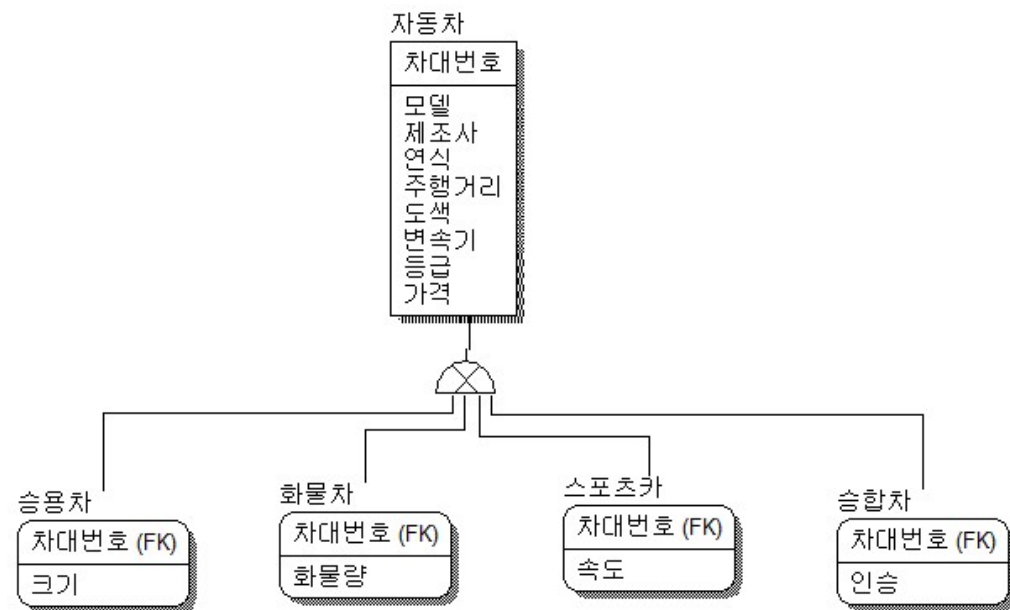
업체는 매물차량을 등록하기 전에 반드시 진단점검을 받아야 한다. 여기서 진단점검은 차량의 상태를 나타내기 위한 최소한의 검사이다.

검사항목으로는 차종확인, 사고여부, 주행진단, 계기판수리/교체흔적, 등록증확인, 차량내부진단, 차량외부진단, 옵션진단, 엔진룸진단, 하부진단이다.

슈퍼타입 작성 이전



슈퍼타입 작성 이후



[예제] (자동차매매) 슈퍼타입/서브타입

회원은 일반구매 회원과 딜러구매 회원으로 나뉜다.

일반구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받는다.

딜러구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받고 추가로 매물목록, 신용등급, 소속회사 연락처를 받는다.

차량은 차대번호, 제조사, 모델, 등급, 연식, 변속기(수동, 자동), 도색, 주행거리, 가격 정보를 갖는다.

차량은 차종에 따라 승용차(소형, 중형, 대형), 화물차(트레일러, 윙바디 등), 승합차(6인승, 9인승, 12인승 등), 스포츠카(300마력, 400마력, 500마력 등)으로 분류할 수 있다.

모든 거래는 저장되며 거래 내용은 구입일시, 남은 보증일수, 게시기간, 회원등급 정보가 기록된다.

업체는 매물차량을 등록하기 전에 반드시 진단점검을 받아야 한다. 여기서 진단점검은 차량의 상태를 나타내기 위한 최소한의 검사이다.

검사항목으로는 차종확인, 사고여부, 주행진단, 계기판수리/교체흔적, 등록증확인, 차량내부진단, 차량외부진단, 옵션진단, 엔진룸진단, 하부진단이다.

일반회원

일반ID
이름
주민등록번호
생년월일
전화번호
핸드폰번호
주소

딜러회원

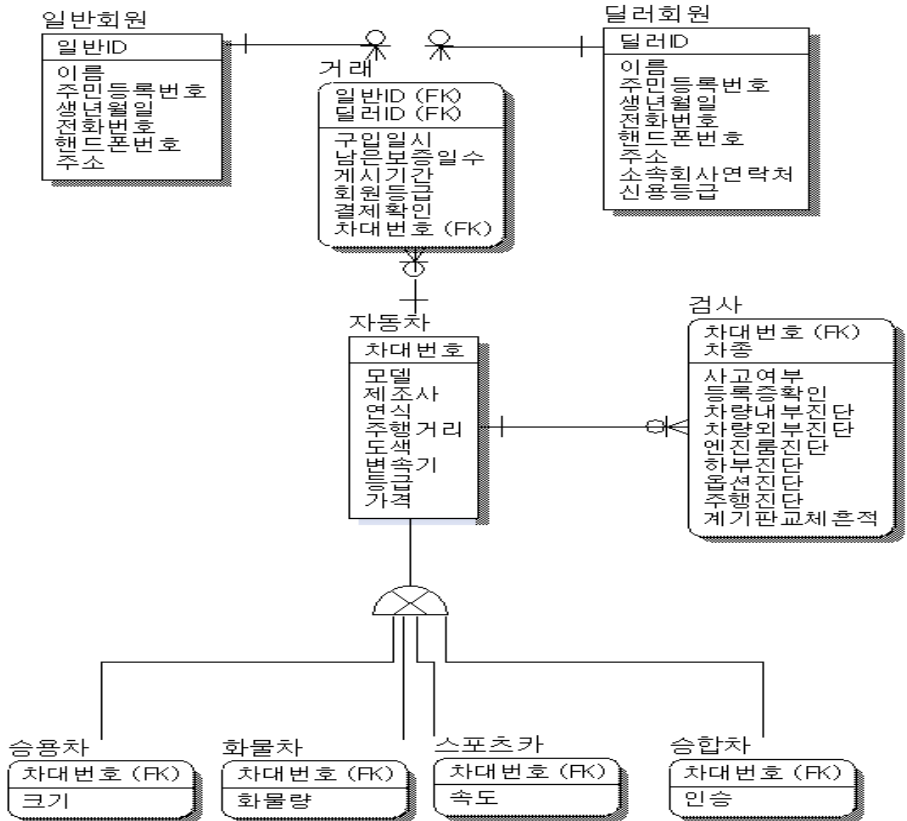
딜러ID
이름
주민등록번호
생년월일
전화번호
핸드폰번호
주소
소속회사연락처
신용등급

자동차

차대번호
모델
제조사
연식
주행거리
도색
변속기
등급
가격

검사

차종
사고여부
등록증확인
인단
차량내부진단
차량외부진단
엔진룸진단
하부진단
옵션진단
주행진단
계기판교체흔적



[예제] (자동차매매) 슈퍼타입/서브타입

회원은 일반구매 회원과 딜러구매 회원으로 나뉜다.

일반구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받는다.

딜러구매 회원은 회원가입시 이름, ID, 생년월일, 주민등록번호, 주소, 전화번호, 핸드폰번호를 입력 받고 추가로 매물목록, 신용등급, 소속회사 연락처를 받는다.

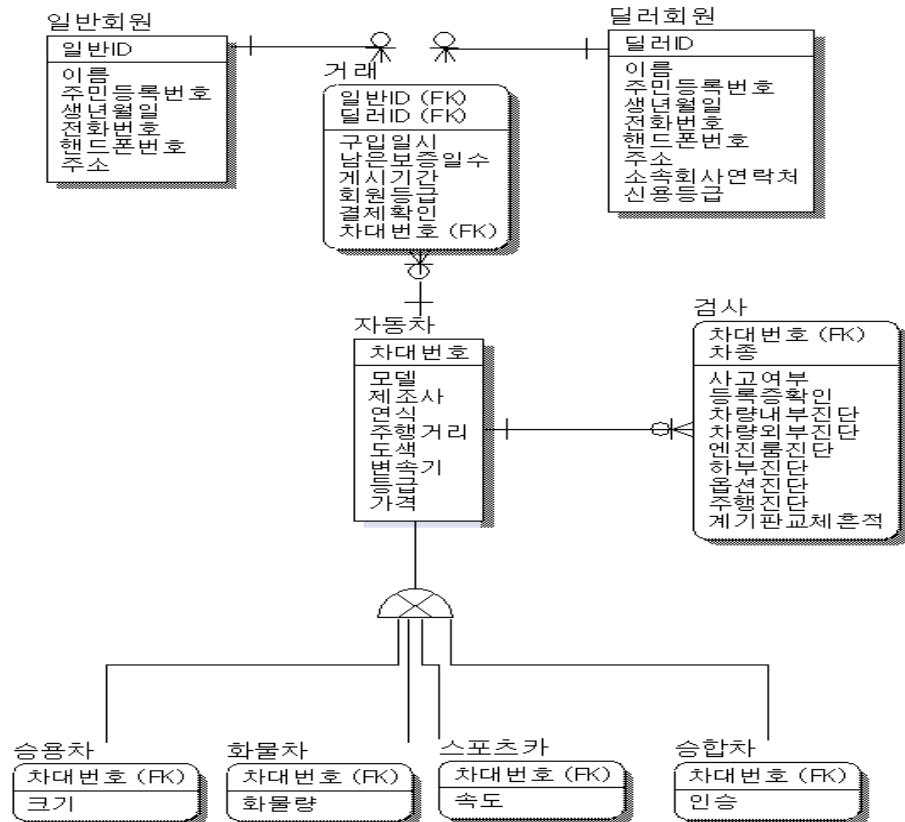
차량은 차대번호, 제조사, 모델, 등급, 연식, 변속기(수동, 자동), 도색, 주행거리, 가격 정보를 갖는다.

차량은 차종에 따라 승용차(소형, 중형, 대형), 화물차(트레일러, 윙바디 등), 승합차(6인승, 9인승, 12인승 등), 스포츠카(300마력, 400마력, 500마력 등)으로 분류할 수 있다.

모든 거래는 저장되며 거래 내용은 구입일시, 남은 보증일수, 게시기간, 회원등급 정보가 기록된다.

업체는 매물차량을 등록하기 전에 반드시 진단점검을 받아야 한다. 여기서 진단점검은 차량의 상태를 나타내기 위한 최소한의 검사이다.

검사항목으로는 차종확인, 사고여부, 주행진단, 계기판수리/교체흔적, 등록증확인, 차량내부진단, 차량외부진단, 옵션진단, 엔진룸진단, 하부진단이다.



<관계스키마>

일반회원(일반ID, 이름, 주민등록번호, 생년월일, 전화번호, 핸드폰번호, 주소)

딜러회원(딜러ID, 이름, 주민등록번호, 생년월일, 전화번호, 핸드폰번호, 주소, 소속회사연락처, 신용등급)

자동차(차대번호, 모델, 제조사, 연식, 주행거리, 도색, 변속기, 등급, 가격)

검사(차대번호, 차종, 사고여부, 등록증확인, 차량내부진단, 차량외부진단, 엔진룸진단, 하부진단, 옵션진단, 주행진단, 계기판교체흔적)

거래(일반ID, 딜러ID, 구입일시, 남은보증일수, 게시기간, 회원등급, 결제확인, 차대번호)

승용차(차대번호, 크기)

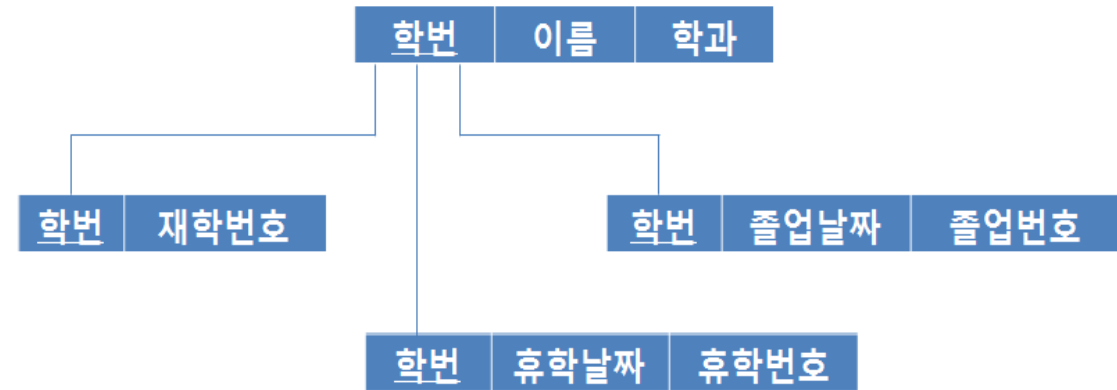
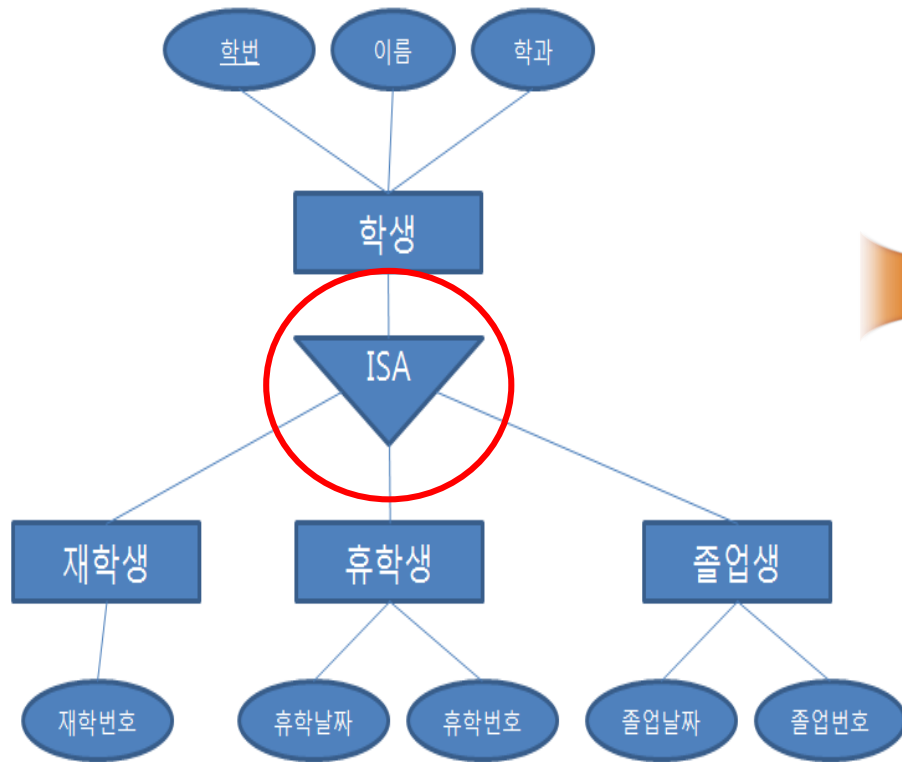
화물차(차대번호, 화물량)

스포츠카(차대번호, 속도)

승합차(차대번호, 인승)

[예제] (학사) 수퍼타입/서브타입

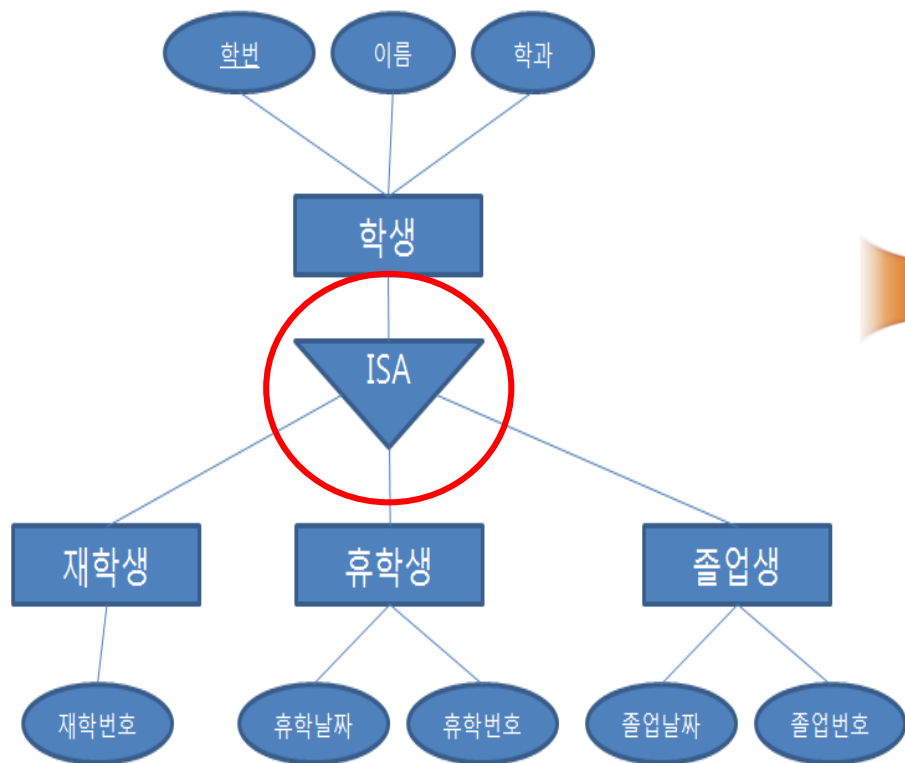
방법 1) 각각의 개체를 모두 릴레이션 스키마로 변환하기



학번	이름	학과	학번	재학번호	
B01	강찬구	생물학	B01	C01	
B02	박찬일	생물학	B02	C02	
B03	유기철	수학	B03	C03	
B04	임창규	수학	B04	C04	
학번	이름	학과	학번	휴학날짜	휴학번호
B05	강홍구	생물학	B05	2010-01-03	D01
B06	김정준	생물학	B06	2003-04-23	D02
B07	김범진	수학	B07	1999-12-24	D03
B08	유 료	수학	B08	2002-07-18	D04
학번	이름	학과	학번	졸업날짜	졸업번호
B09	최성은	생물학	B09	1999-02-24	E01
B10	원승호	생물학	B10	1001-08-24	E02
B11	반현오	수학	B11	2006-02-24	E03
B12	이오십	수학	B12	2009-02-24	E04

[예제] (학사) 수퍼타입/서브타입

방법 2) 각각의 하위 개체를 릴레이션 스키마로 변환하기



<재학생>

학번	이름	학과	재학번호
B01	강찬구	생물학	C01
B02	박찬일	생물학	C02
B03	유기철	수학	C03
B04	임창규	수학	C04

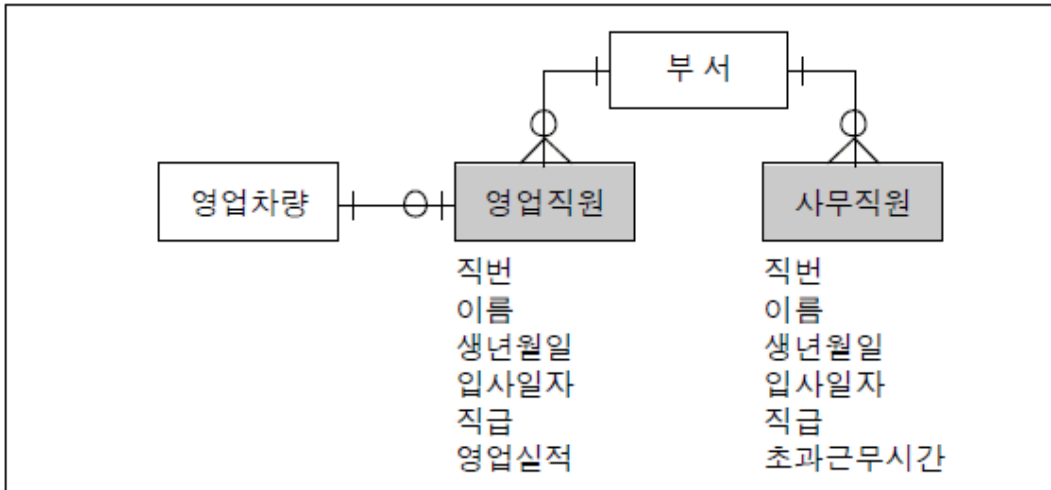
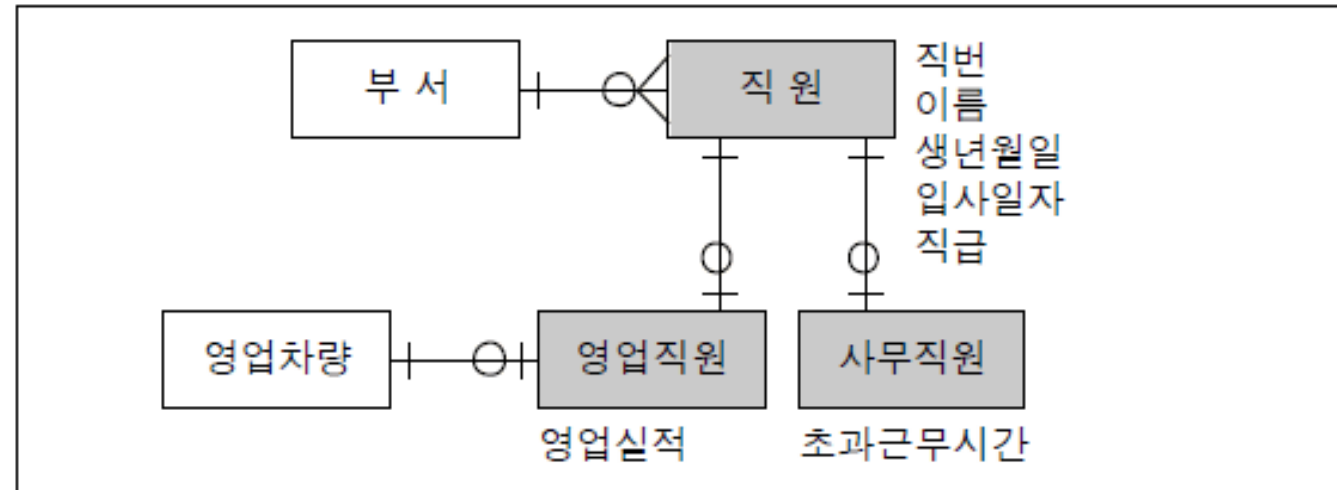
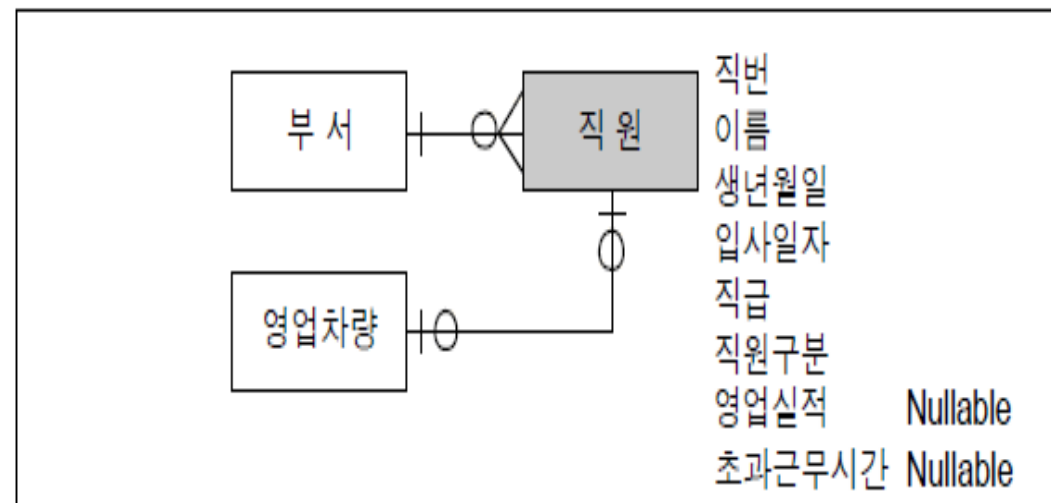
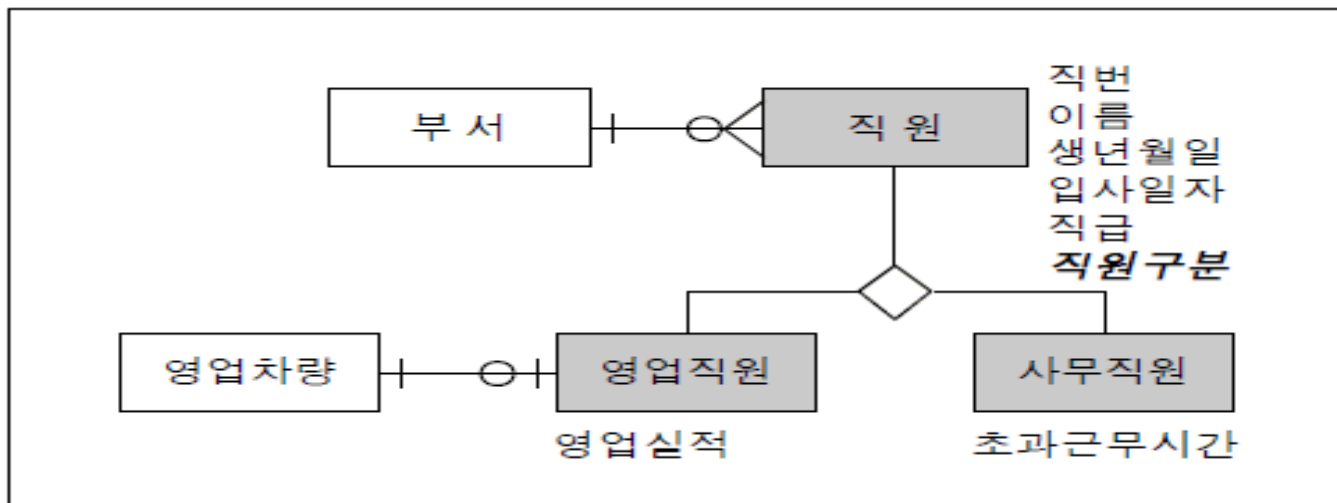
<휴학생>

학번	이름	학과	휴학날짜	휴학번호
B05	강흥구	생물학	2010-01-03	D01
B06	김정준	생물학	2003-04-23	D02
B07	김범진	수학	1999-12-24	D03
B08	유 료	수학	2002-07-18	D04

<졸업생>

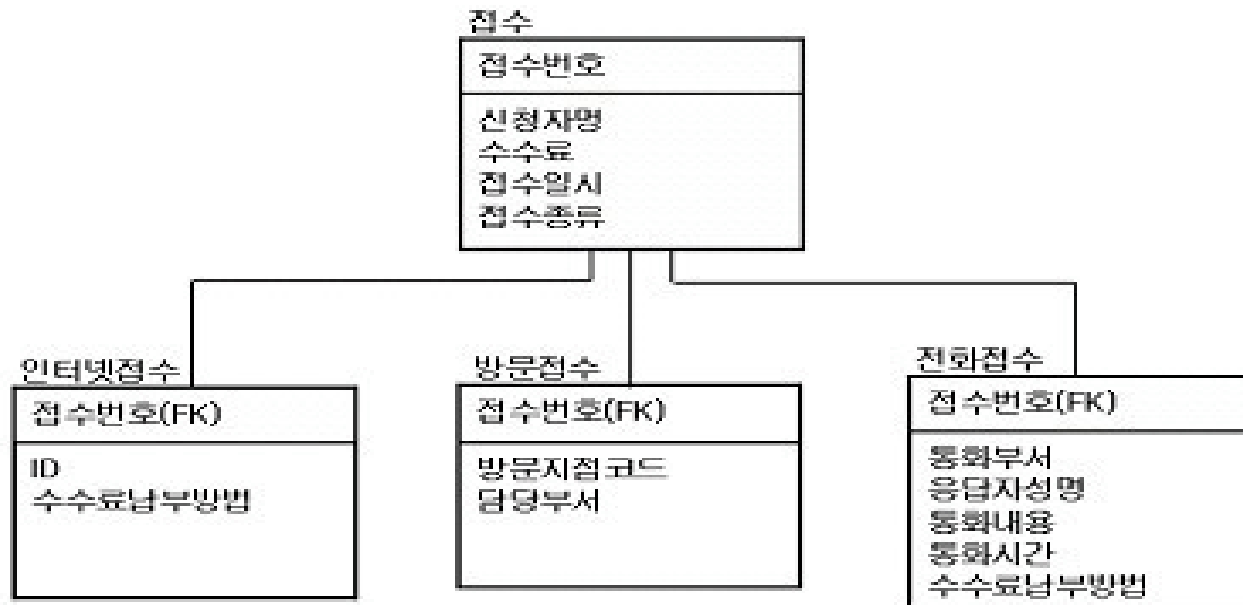
학번	이름	학과	졸업날짜	졸업번호
B09	최성은	생물학	1999-02-24	E01
B10	원승호	생물학	1001-08-24	E02
B11	반현오	수학	2006-02-24	E03
B12	이오십	수학	2009-02-24	E04

[예제] (사원) 수퍼타입/서브타입



[Quiz 11-03] Erwin 논리적 설계(수퍼/서브)

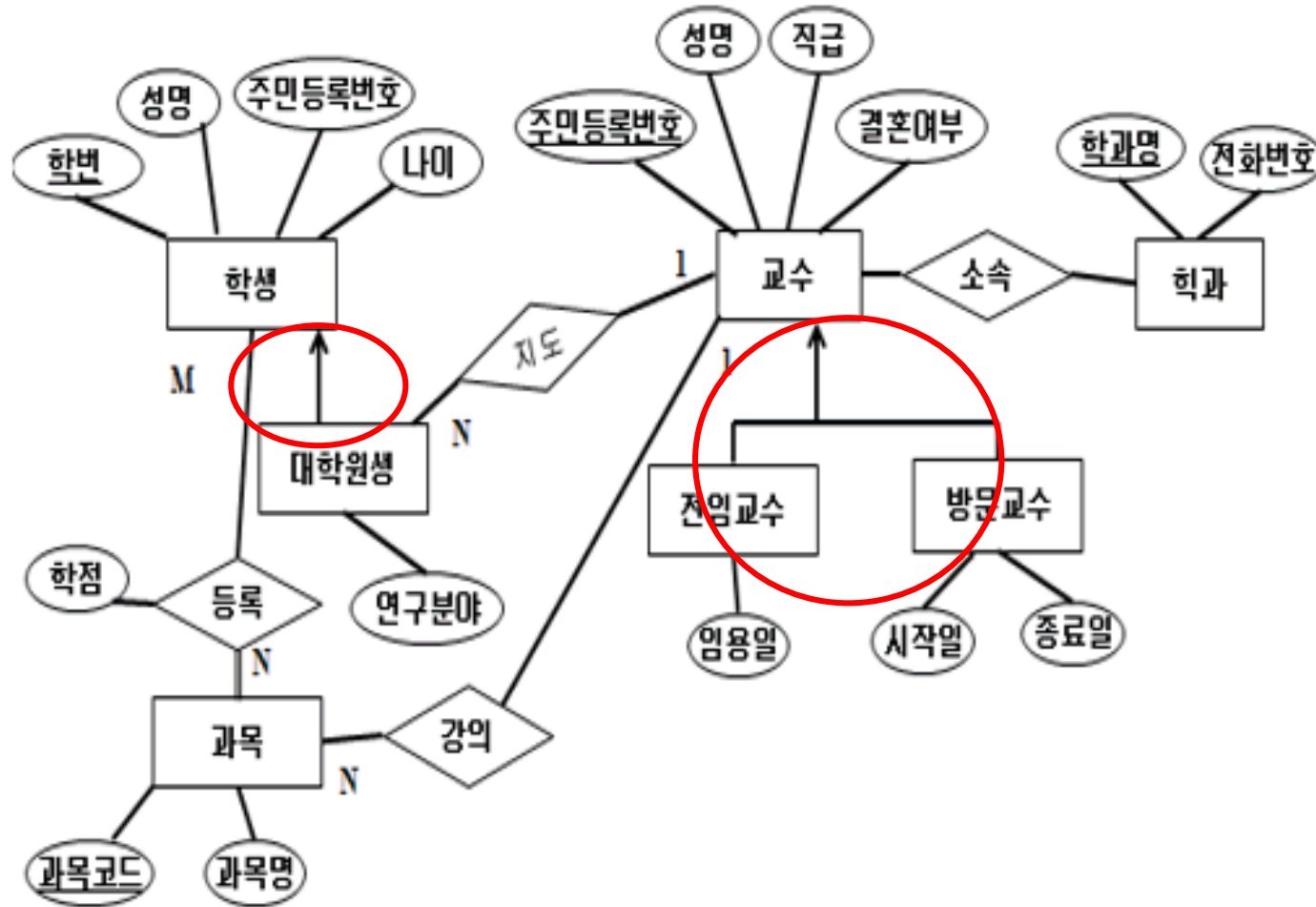
수퍼/서브 타입을 사용하여 ERWin으로 모델링하시오



- 1) 3개의 엔티티로 변환
- 2) 2개의 엔티티로 변환
- 3) 1개의 엔티티로 변환

[Quiz 11-04] Erwin 논리적 설계(수퍼/서브)

수퍼/서브 타입을 사용하여 ERWin으로 모델링하시오

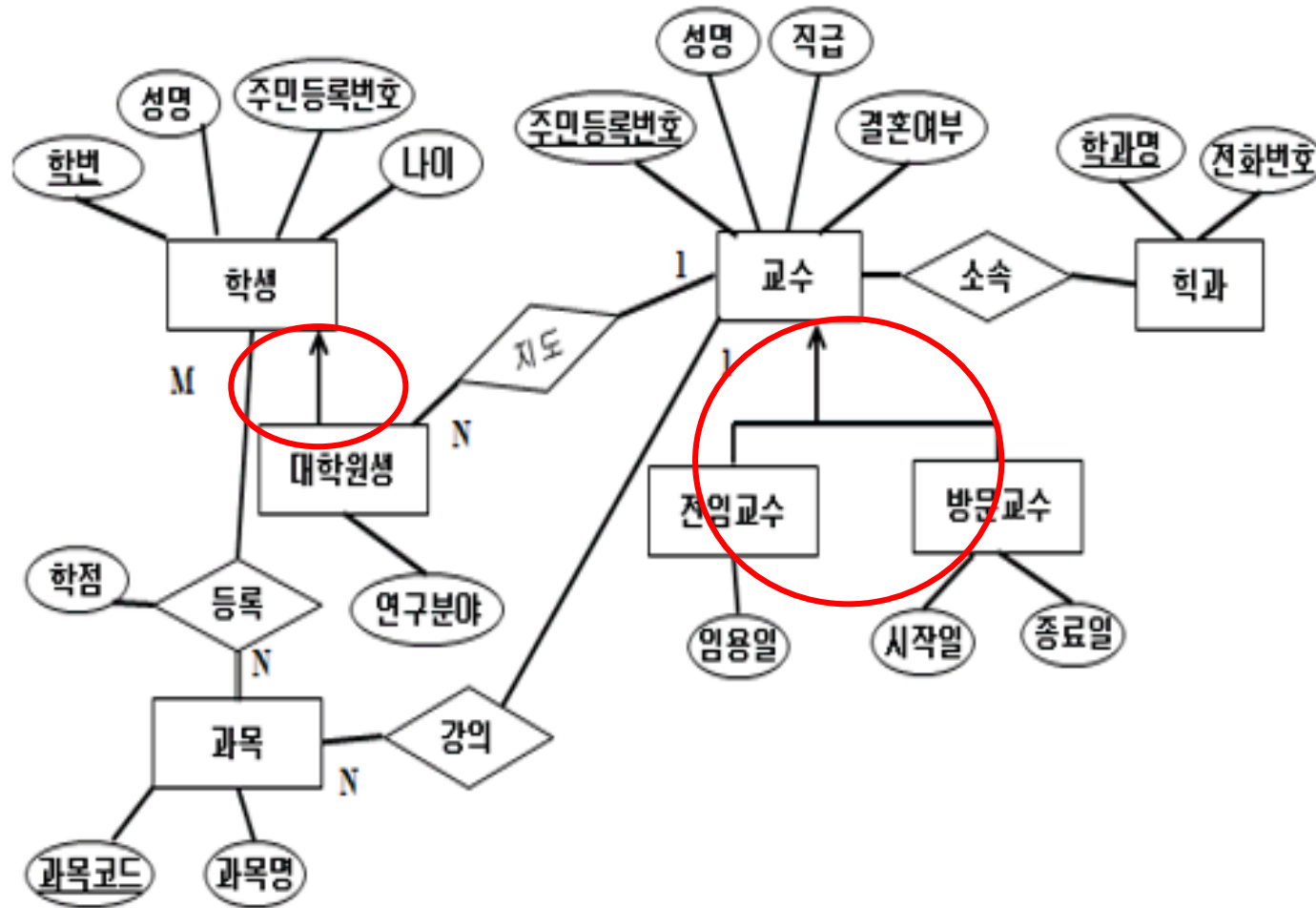


문제 해설

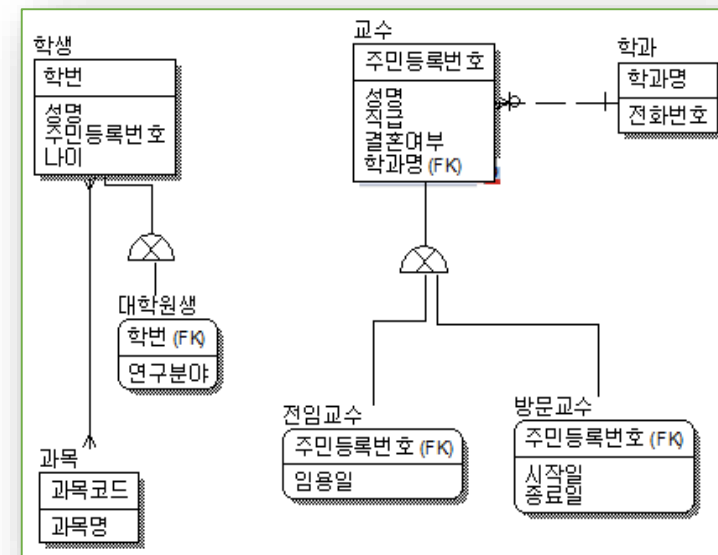
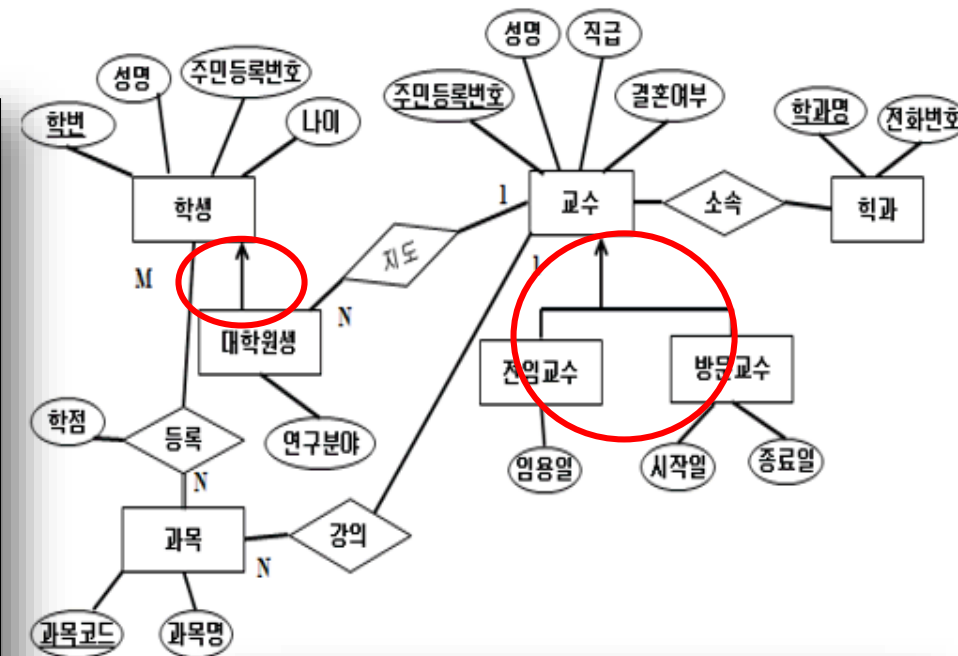
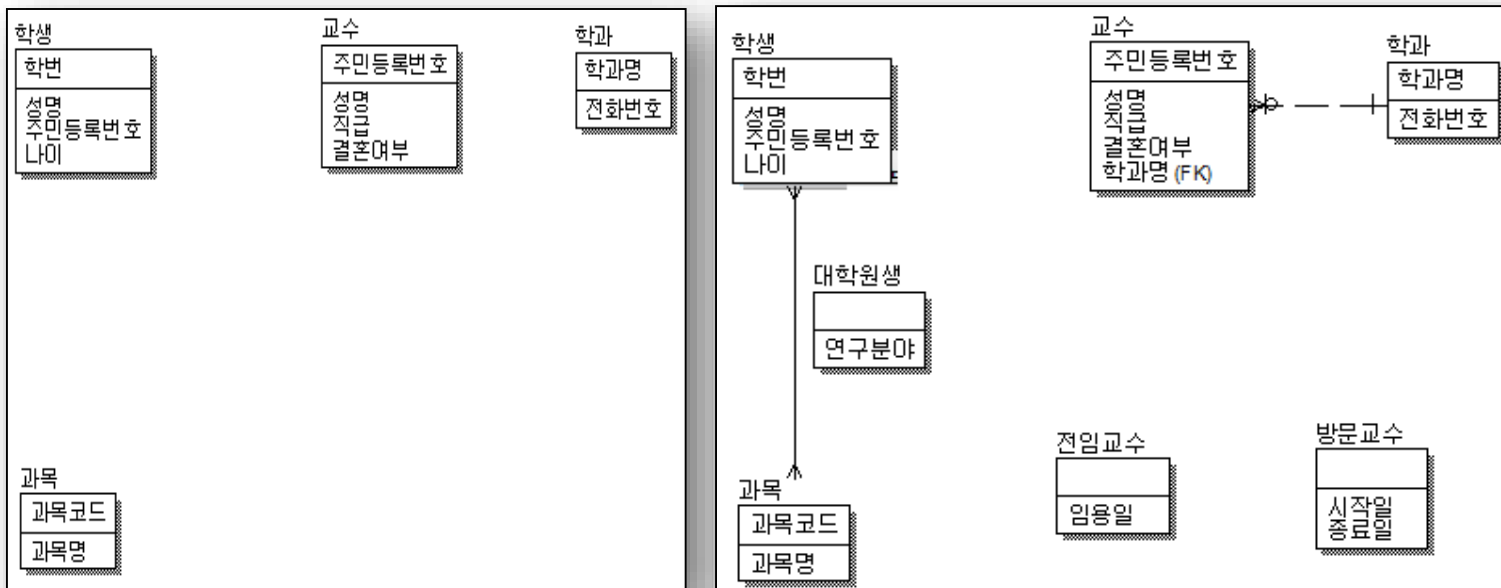
담당교수: 김희숙
(jasmin11@hanmail.net)

[해설][Quiz 11-04] Erwin 논리적 설계(수퍼/서브)

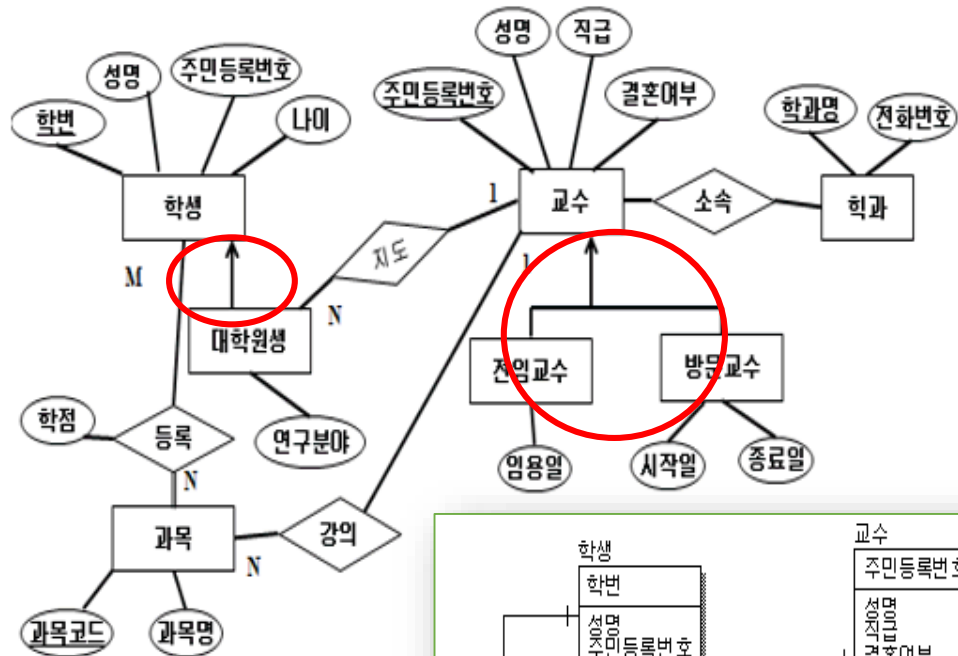
수퍼/서브 타입을 사용하여 ERWin으로 모델링하시오



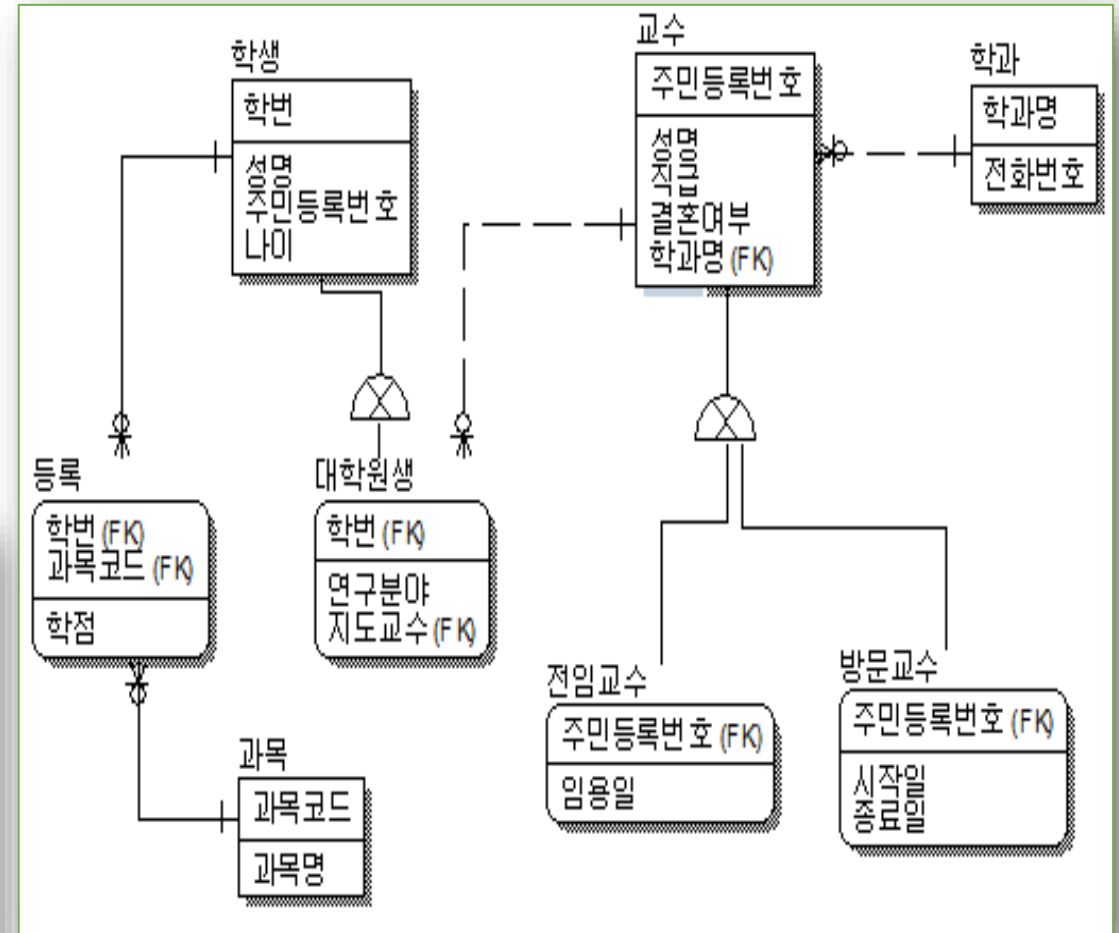
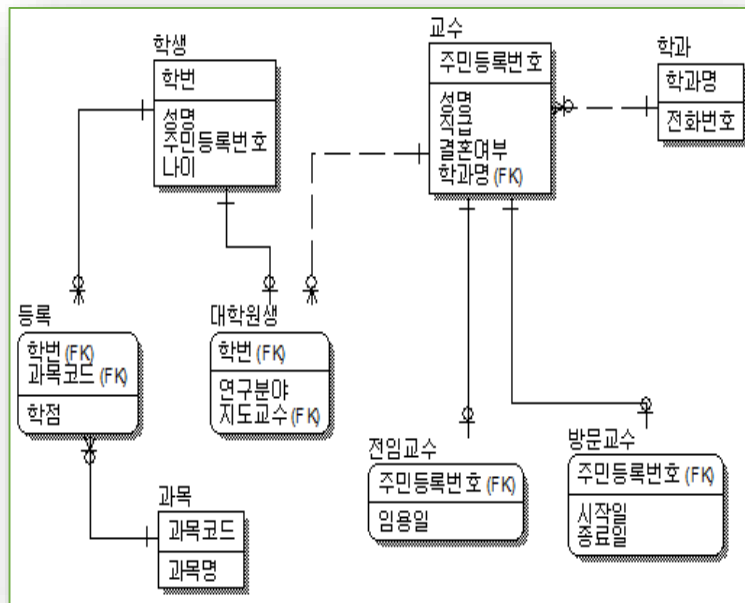
수퍼/서브 타입을 사용하여 ERWin으로 모델링하시오



수퍼/서브 타입을 사용하여 ERWin으로 모델링하시오



<최종>



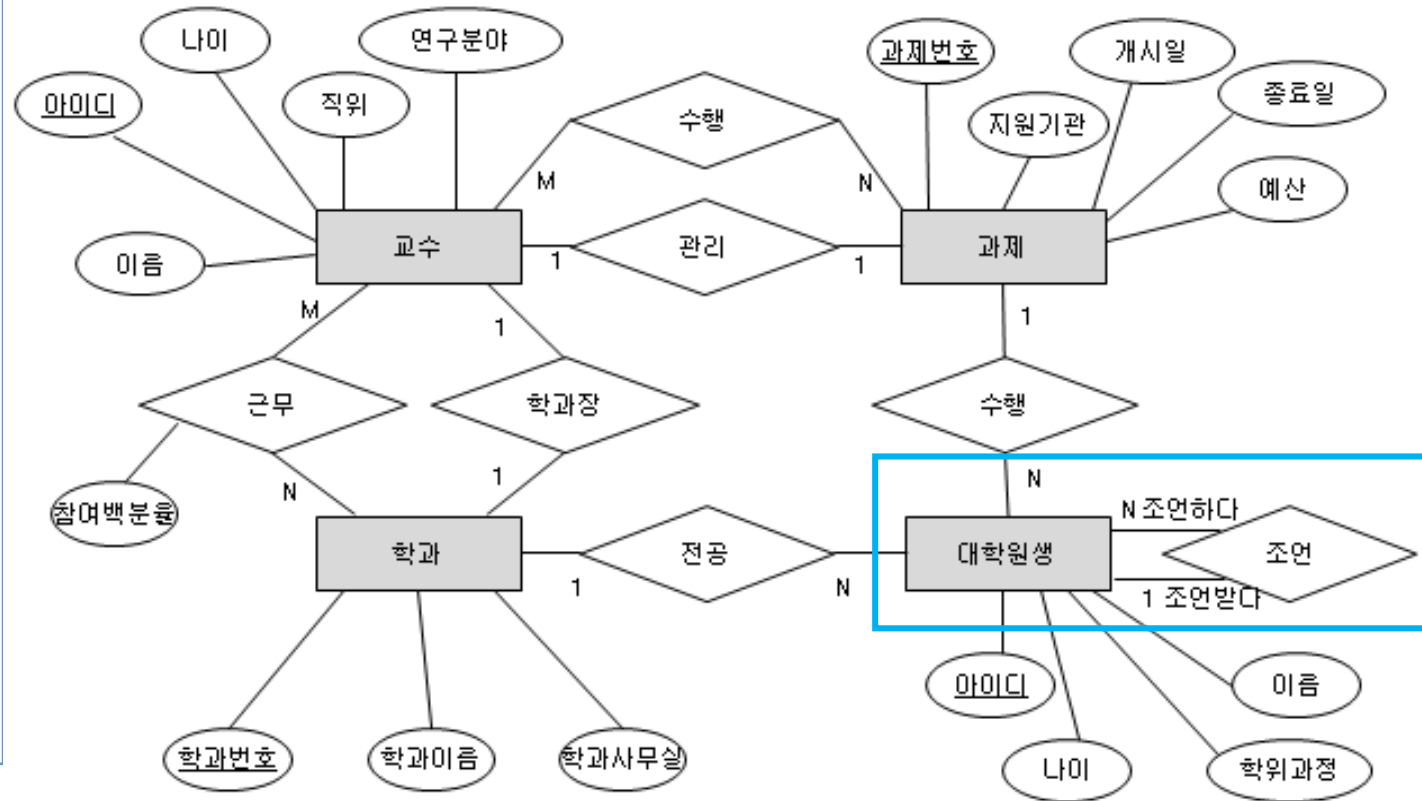
수퍼/서브 타입을 identity 로 전환

[해설][Quiz 11-05] Erwin 논리적 설계

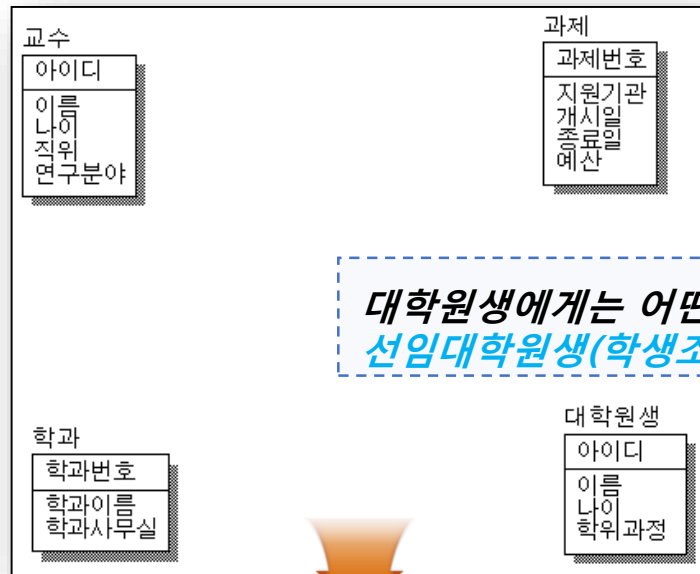
교수는 아이디, 이름, 나이, 직위, 연구분야를 가진다
 학과에는 학과번호, 학과이름, 학과사무실이 있다
 대학원생은 아이디, 이름, 나이, 학위과정(석사/박사)을 가진다
 과제는 과제번호, 지원기관, 개시일, 종료일, 예산액이 있다

학과마다 그 학과를 운영하는 교수(학과장이라고 한다)가 한 명씩 있다
 한 교수가 여러 학과에서 근무할 수 있는데, 이때 각 학과별로 참여백분율이 기록된다
 대학원생에게는 학위과정을 밟을 전공학과가 하나씩 있다
 대학원생에게는 어떤 과목을 들으면 좋을지 조언해주는 **선임대학원생(학생조언자)**이 있다
 과제는 한 교수(연구책임자라고 한다)에 의해 관리된다
 과제는 한 사람 이상의 교수(공동연구책임자라고 한다)에 의해 수행된다
 한 과제는 한 명 이상의 대학원생(연구조교라고 한다)에 의해 수행된다

<마당대학 ERD>

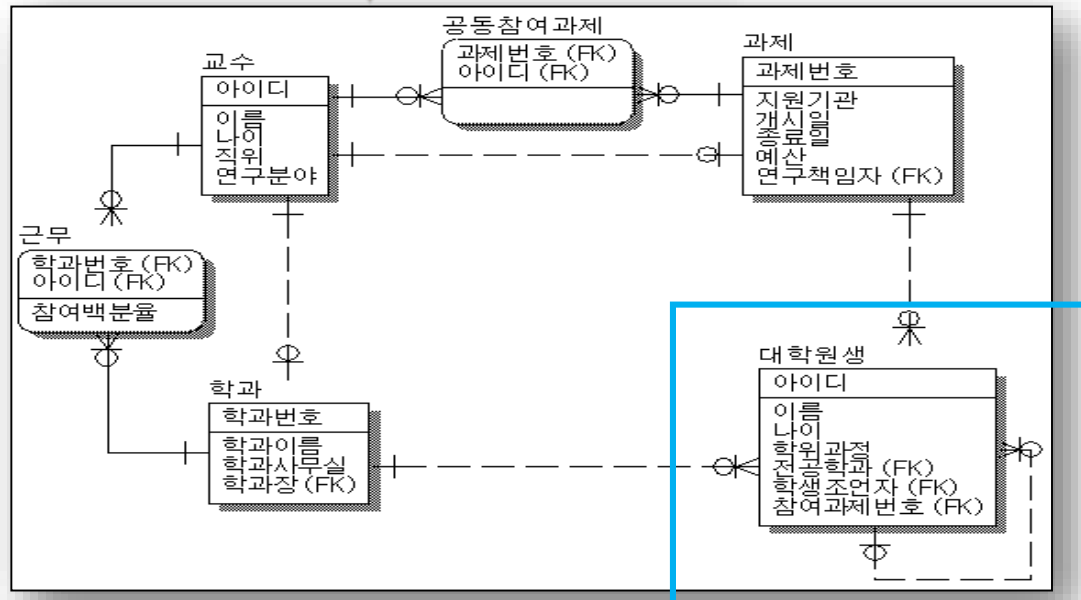
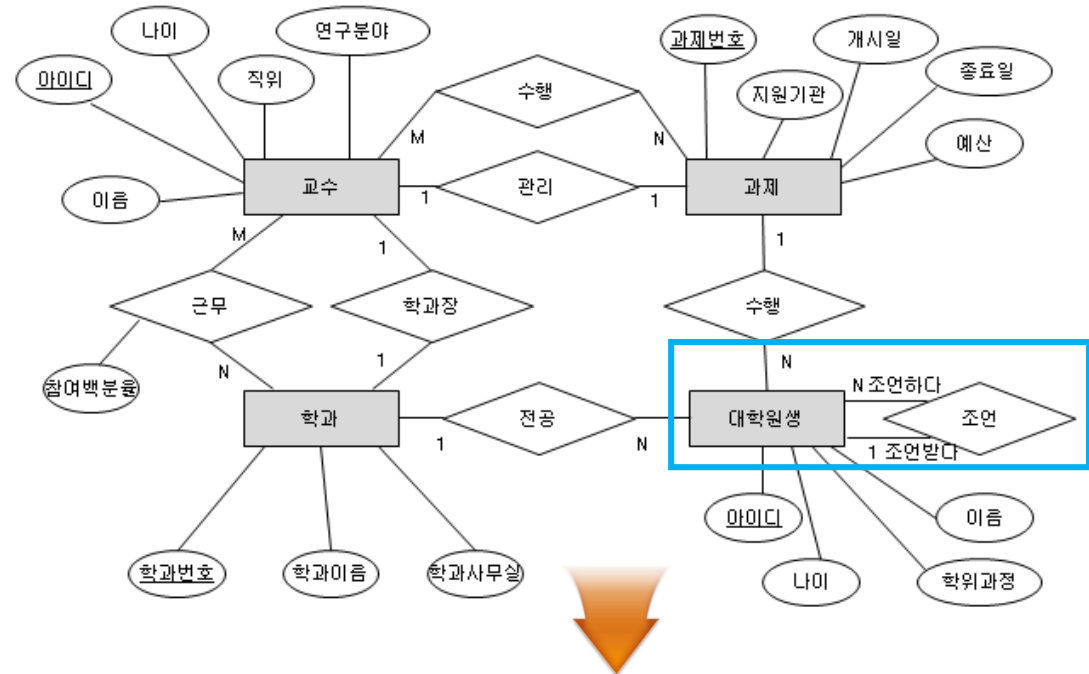


[예제] [해설] [Quiz 11-05] Erwin 논리적 설계(재귀적 관계)



대학원생에게는 어떤 과목을 들으면 좋을지 조언해주는
선임대학원생(학생조연자)이 있다

<마당대학 ERD>



<관계 스키마>

교수(아이디 pk, 이름, 나이, 직위, 연구분야)
 학과(학과번호 pk, 학과이름, 학과사무실, 학과장(FK))
 과제(과제번호 pk, 지원기관, 개시일, 종료일, 예산, 연구책임자(FK))
 대학원생(아이디 pk, 이름, 나이, 학위과정, 전공학과(FK),
학생조연자(FK), 참여과제번호(FK))
 공동참여과제(참여과제번호 pk(FK), 공동연구책임자 pk(FK))
 근무(학과번호 pk(FK), 교수아이디 pk(FK), 참여백분율)

[예제 12-01] 정규화

다음 일일재고를 정규화하시오

일일재고

물류센터코드 재고일자			
월초재고수량			
장기재고1개월수량			
장기재고2개월수량			
장기재고3개월수량			
장기재고1개월주문수량			
장기재고2개월주문수량			
장기재고3개월주문수량			
장기재고1개월금액			
장기재고2개월금액			
장기재고3개월금액			
장기재고1개월주문금액			
장기재고2개월주문금액			
장기재고3개월주문금액			

❖ 정규화

- 1) 제1정규형(1NF)
- 2) 제2정규형(2NF)
- 3) 제3정규형(3NF)

[해설][예제 12-01] 정규화

다음 일일재고를 정규화하시오

일일재고

물류센터코드 재고일자			
월초재고수량			
장기재고1개월수량	수량		
장기재고2개월수량	수량		
장기재고3개월수량	수량		
장기재고1개월주문수량	수량		
장기재고2개월주문수량	수량		
장기재고3개월주문수량	수량		
장기재고1개월주문금액	금액		
장기재고2개월주문금액	금액		
장기재고3개월주문금액	금액		



일일재고

물류센터코드 재고일자
월초재고수량

일일재고상세

물류센터코드 (FK) 재고일자 (FK) 재고기간
장기재고수량
장기재고주문수량
장기재고주문금액
장기재고주문금액

❖ 정규화

1) 제1정규형(1NF)

2) 제2정규형(2NF)

3) 제3정규형(3NF)

[해설][예제 12-02] 결제방식

-결제
결제유형 (0:신용카드,
1:무통장입금,
2:휴대폰)

결제
결제번호
결제자성명
연락처
결제유형

신용카드
결제번호 (FK)
신용카드번호
유효기간
결제일자

무통장입금
결제번호 (FK)
은행명
계좌번호
입금일자

휴대폰결제
결제번호 (FK)
휴대폰번호



-결제
결제유형 (0:신용카드,
1:무통장입금,
2:휴대폰)

수퍼/서브 타입을 identity 로 전환

결제
결제번호
결제자성명
연락처
결제유형

신용카드
결제번호 (FK)
신용카드번호
유효기간
결제일자

무통장입금
결제번호 (FK)
은행명
계좌번호
입금일자

휴대폰결제
결제번호 (FK)
휴대폰번호

장점: 데이터 반복을 최소화

- 1) 중복값이 없으므로 이상현상 적다
- 2) 데이터 무결성 보장

단점: 데이터들이 많이 분리

- 1) 성능저하

[예제 12-03] 항공기 좌석예약

-항공기 좌석예약

항공사에서 회원 관리, 비행기 관리, 좌석 예약 관리를 위한 데이터베이스를 구축하고자 한다.

항공사에 회원으로 가입하려면 회원아이디, 비밀번호, 성명, 신용카드 정보를 입력해야 한다.

회원의 신용카드 정보는 여러 개를 저장할 수 있는데, 세부적으로는 신용카드번호, 유효기간을 저장할 수 있다.

항공사에서는 보유한 비행기에 대해 비행기번호, 출발날짜, 출발시간 정보를 저장하고 있다.

항공사에서는 좌석에 대해 좌석번호, 등급 정보를 저장하고 있다.

회원은 좌석을 예약하는데, 회원 한 명은 좌석을 하나만 예약할 수 있고, 한 좌석은 회원 한 명만 예약할 수 있다.

비행기에는 좌석이 존재하는데, 비행기 하나에는 좌석이 여러 개 존재할 수 있고 한 좌석은 반드시 하나의 비행기에만 존재해야 한다.

그리고 좌석은 비행기가 없으면 의미가 없다.

[해설][예제 12-03] 항공기 좌석예약

항공사에 **회원**으로 가입하려면 **회원아이디**, **비밀번호**, **성명**, **신용카드 정보**를 입력해야 한다.

회원의 신용카드 정보는 여러 개를 저장할 수 있는데, 세부적으로는 **신용카드번호**, **유효기간**을 저장할 수 있다.

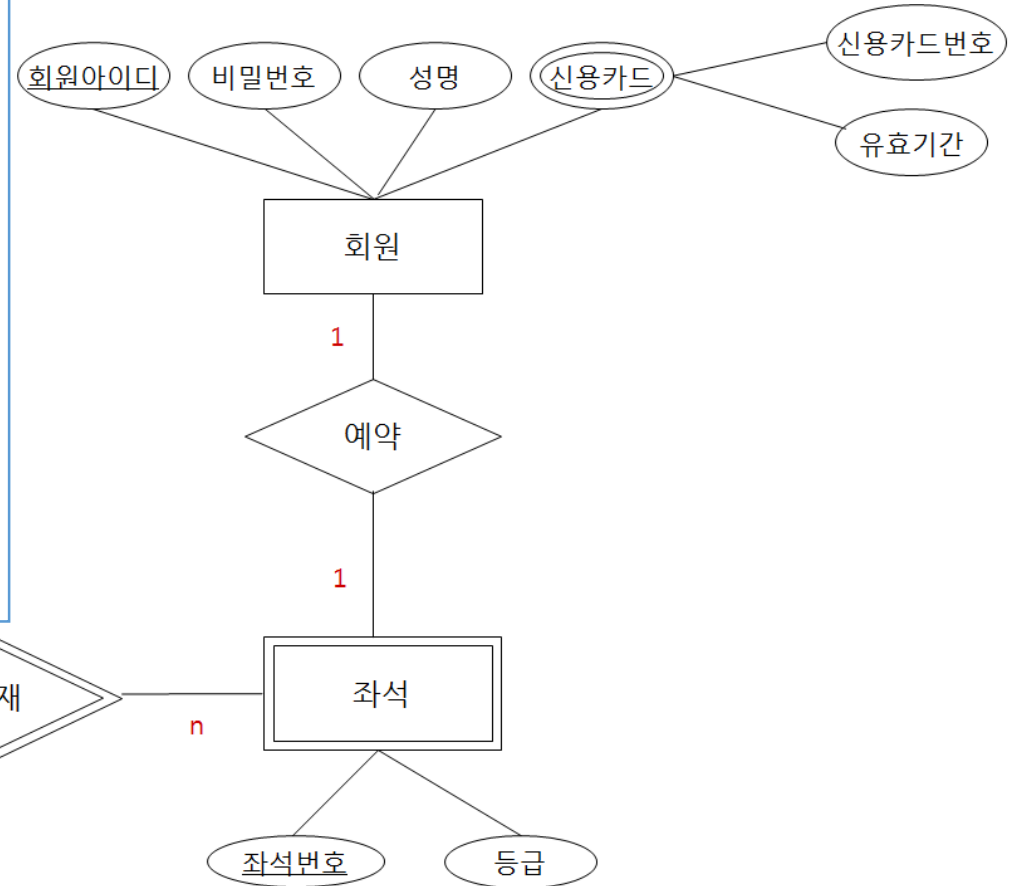
항공사에서는 보유한 **비행기**에 대해 **비행기번호**, **출발날짜**, **출발시간** 정보를 저장하고 있다.

항공사에서는 **좌석**에 대해 **좌석번호**, **등급** 정보를 저장하고 있다.

회원은 좌석을 **예약**하는데, 회원 한 명은 좌석을 하나만 예약할 수 있고, 한 좌석은 회원 한 명만 예약할 수 있다.

비행기에는 좌석이 **존재**하는데, 비행기 하나에는 좌석이 여러 개 존재할 수 있고 한 좌석은 반드시 하나의 비행기에만 존재해야 한다.

그리고 좌석은 비행기가 없으면 의미가 없다.

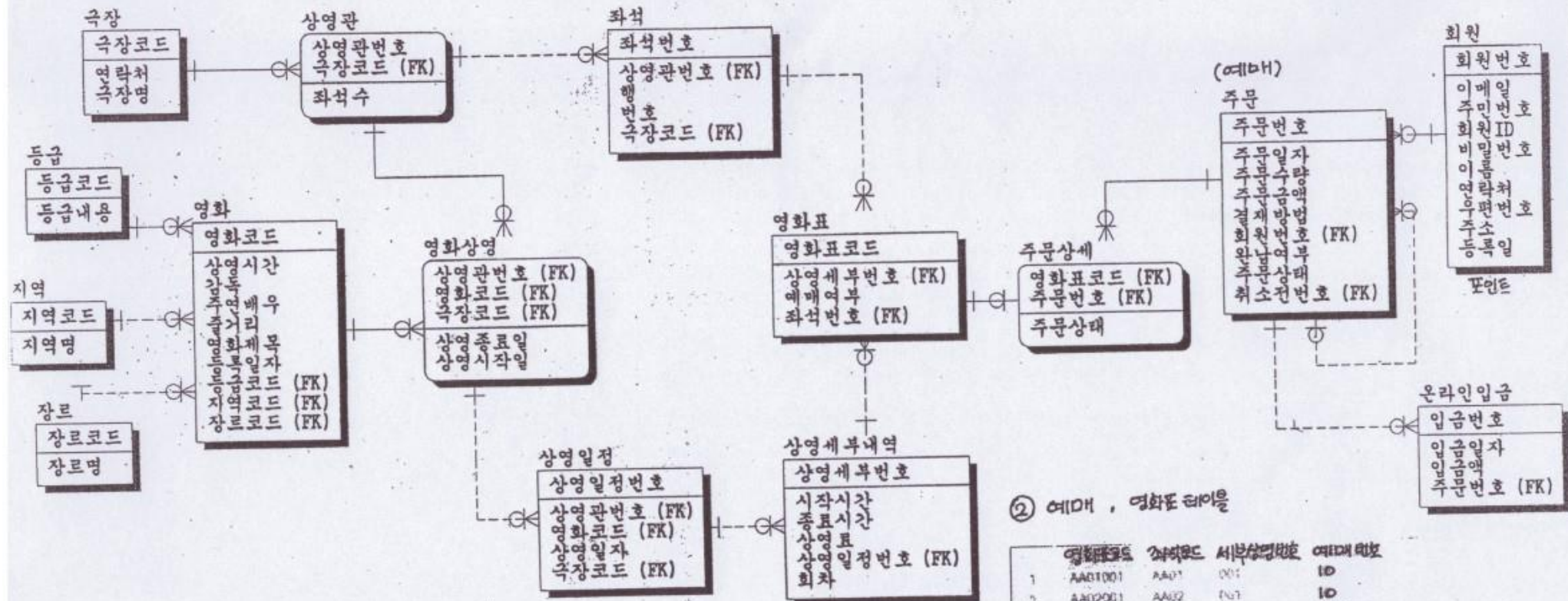


1) 엔티티(마스터 테이블)

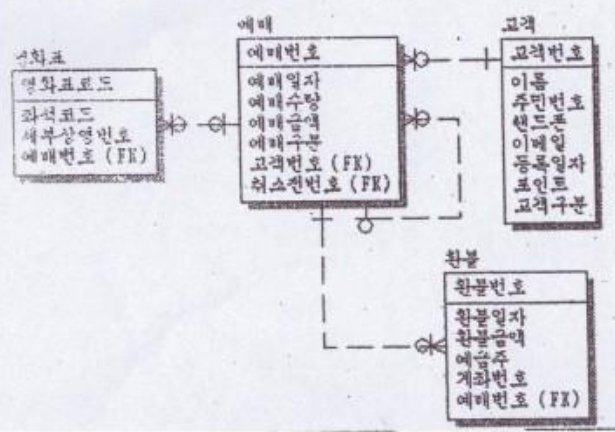
회원, 비행기, 좌석

2) 관계 테이블

예약, 존재



예: <영화표, 예매 테이블>



① 영화표 코드 생성 : ex) AA01001 좌석코드 시무상영번호

영화표

영화표코드	좌석코드	시무상영번호	예매번호
AA01001	AA01	001	NULL
AA02001	AA02	001	NULL
AA03001	AA03	001	NULL
AA04001	AA04	001	NULL
AA05001	AA05	001	NULL
AA06001	AA06	001	NULL
AA07001	AA07	001	NULL

② 예매, 영화표 테이블

영화표코드	좌석코드	시무상영번호	예매번호
1 AA01001	AA01	001	10
2 AA02001	AA02	001	10
3 AA03001	AA03	001	NULL
4 AA04001	AA04	001	10
5 AA05001	AA05	001	NULL
6 AA06001	AA06	001	NULL
7 AA07001	AA07	001	NULL

예매 테이블
예매번호 예매일자 예매수량 예매금액 예매구분 예매구분 코드번호 취소금액

1	10	20110506	3	21000	1	100	NULL
---	----	----------	---	-------	---	-----	------

③ 예매취소 후,

영화표코드	좌석코드	시무상영번호	예매번호
1 AA01001	AA01	001	10
2 AA02001	AA02	001	10
3 AA03001	AA03	001	NULL
4 AA04001	AA04	001	NULL
5 AA05001	AA05	001	NULL
6 AA06001	AA06	001	NULL
7 AA07001	AA07	001	NULL

예매 테이블
예매번호 예매일자 예매수량 예매금액 예매구분 예매구분 코드번호 취소금액

1	10	20110506	3	21000	1	100	NULL
2	11	20110506	2	14000	2	100	10

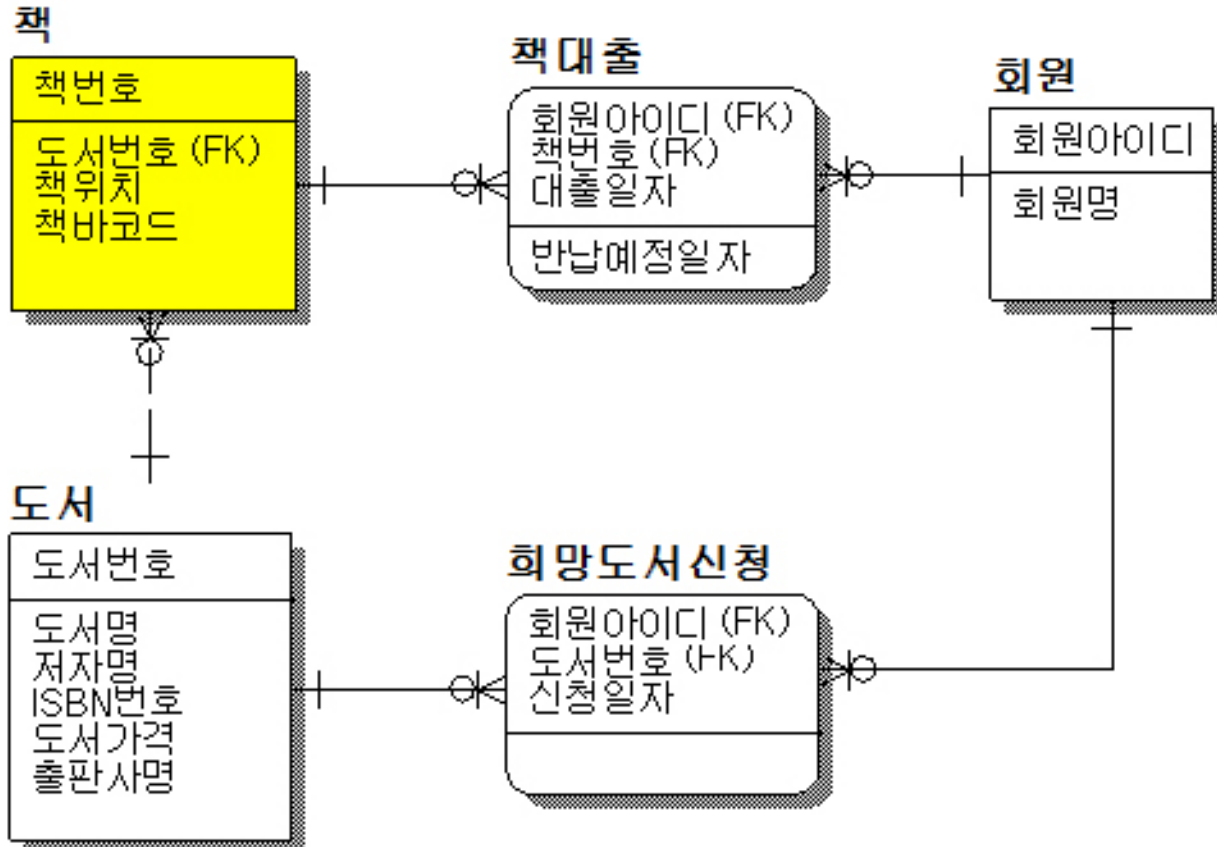
실체 엔티티/종속 엔티티

12주차 2-01

담당교수: 김희숙
(jasmin11@hanmail.net)

[예제] 실체 엔티티

❖ **실체 엔티티**: 실제로 보이는 것을 나타내는 데이터를 관리하는 엔티티



도서관 업무 설계 모델:

책을 대출할 수 있고 희망도서를 신청할 수 있는 간략한 모델

1) **도서관에 있는 책**: 실체 엔티티인가?

대출: 실제 책을 손에 들고 빌린다

2) **희망도서**: 실체 엔티티인가?

책을 나타내는 정보를 지목해서 구매요청

희망도서를 신청할 때는 실제 책이 아니라 책의 개념을 신청

1) **책**: 실체 엔티티

날권의 실제 책(인스턴스 책 개수만큼 여러 개 존재)

2) **도서**: 실체 엔티티 아니다

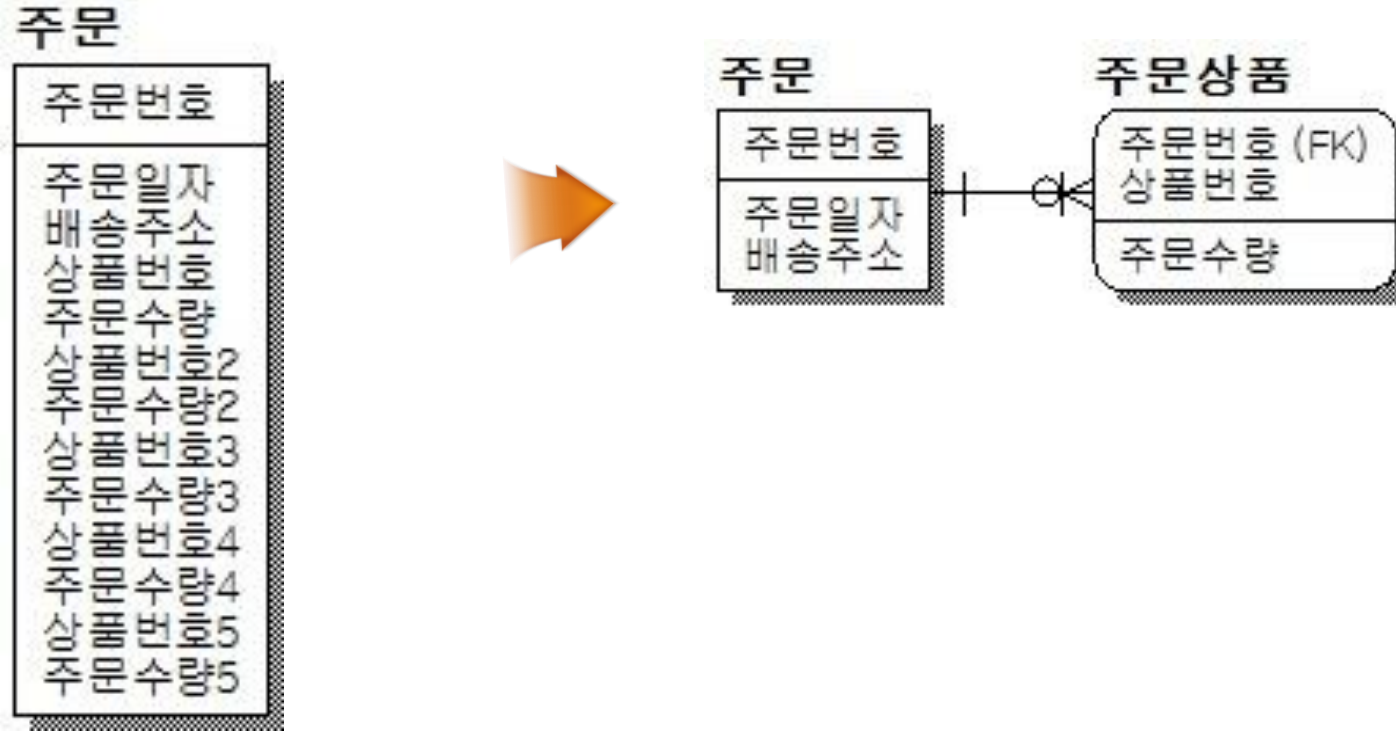
책 정보(인스턴스 한 개만 존재)

(예) 간단한 도서관 모델



[예제] 종속 엔티티

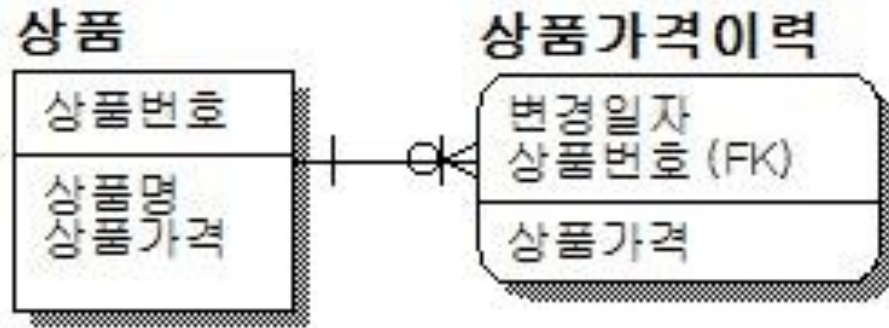
❖ **종속 엔티티**: 상위(부모) 엔티티가 없으면 존재할 수 없는 엔티티



(예) 주문상품: 주문 엔티티에 존재종속

[예제] 종속 엔티티

❖ **종속 엔티티**: 상위(부모) 엔티티가 없으면 존재할 수 없는 엔티티



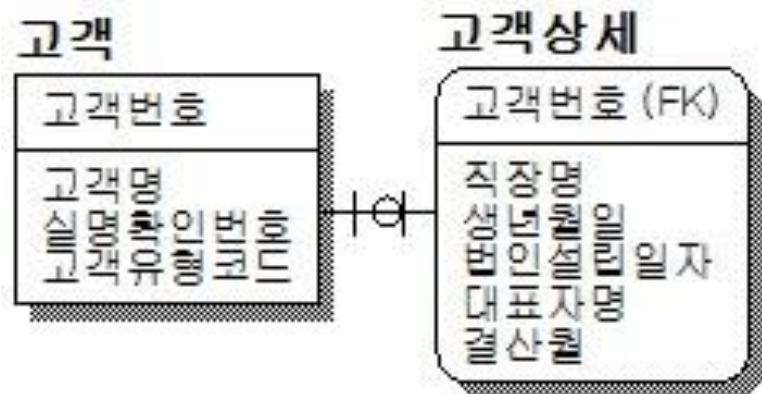
(예) 상품가격이력 엔티티

상품 엔티티의 상품가격 속성 값이 변경됐을 때,
변경되기 전의 상품가격 값을 관리하는 엔티티

원천 데이터인 상품 엔티티가 없다면 변경이력 데이터는 존재할 수 없다

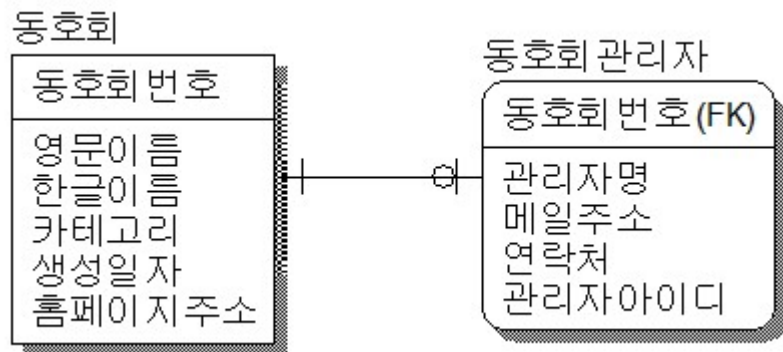
[예제] 종속 엔티티

❖ **종속 엔티티**: 상위(부모) 엔티티가 없으면 존재할 수 없는 엔티티



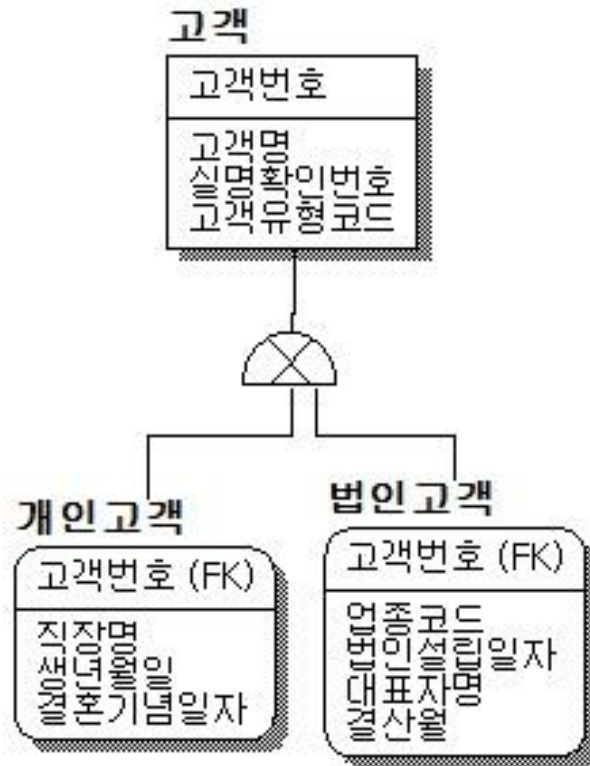
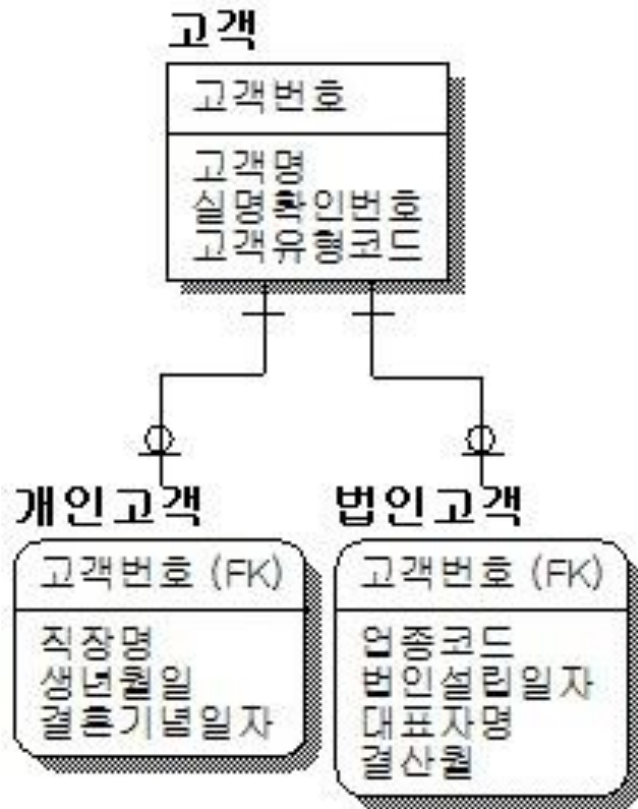
예)

하나의 엔터티인데 분해해서 일대일(1:1) 관계의 엔터티로 설계한 모델



[예제] 종속 엔티티

❖ **종속 엔티티**: 상위(부모) 엔터티가 없으면 존재할 수 없는 엔터티



예) 서브타입 엔티티

하나의 수퍼타입 엔터티를 일대일(1:1) 관계의 엔터티로 분리한 것이기 때문에 존재 종속된 엔터티

[예제] 통합

통합계좌알림서비스

통합계좌번호
신청일자
신입종류코드
알림방법코드
알림반복횟수

계좌알림서비스

통합계좌번호
계좌순번
신청일자
신입종류코드
알림방법코드
알림반복횟수



계좌통합알림서비스

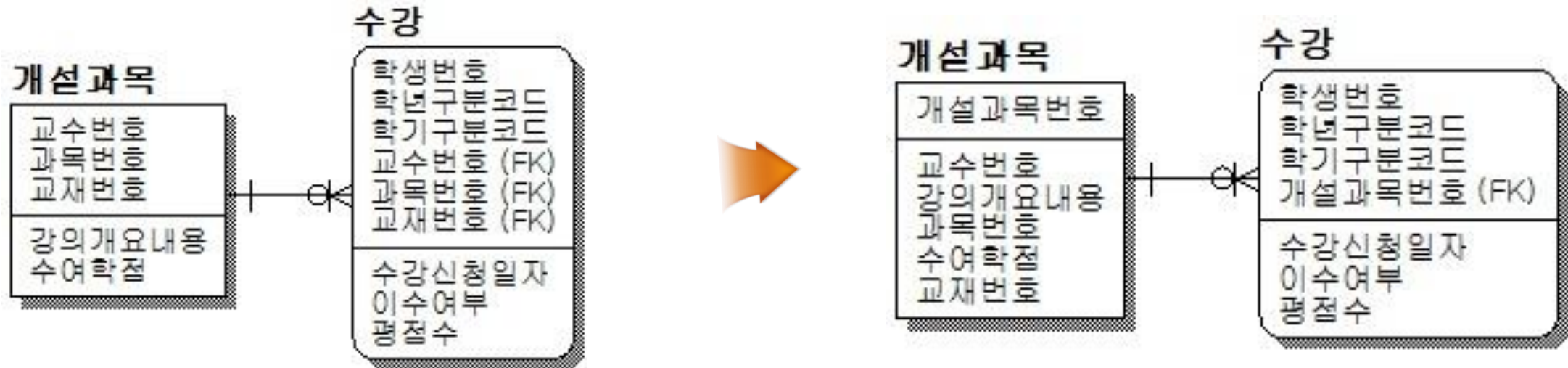
통합계좌번호
계좌순번
계좌구분코드
신청일자
신입종류코드
알림방법코드
알림반복횟수

예) 통합대상 알림서비스 엔터티

장점: 계좌 알림서비스는 계좌통합알림서비스 엔터티에서 관리가능
(주 식별자가 같기 때문에 가능)

단점: 계좌통합알림서비스 엔터티에서
통합계좌알림서비스를 관리하려면 계좌순번 속성이 문제가 된다
통합계좌에 대해서는 계좌순번 속성에 값이 없기 때문
(이 때 '00' 등의 기본값 사용)=>논리적인 널 값 사용
계좌구분코드 속성값에는 '통합계좌'로 관리

[예제] 인조식별자



예) 업무식별자를 주식별자로 사용한 경우

[예제] 인조식별자

주문

고객번호	주문일자
상품번호	주문수량
주문금액	전화번호
연락처	주소
배송지	



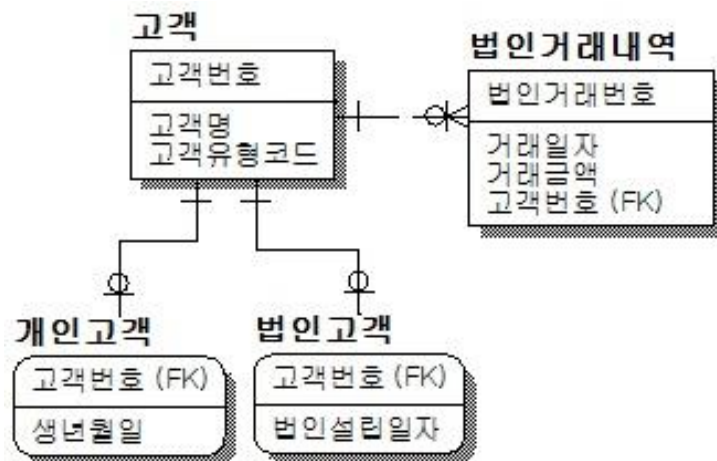
주문

주문번호
고객번호
주문수량
상품번호
주문금액
연락처
전화번호
주소
배송지

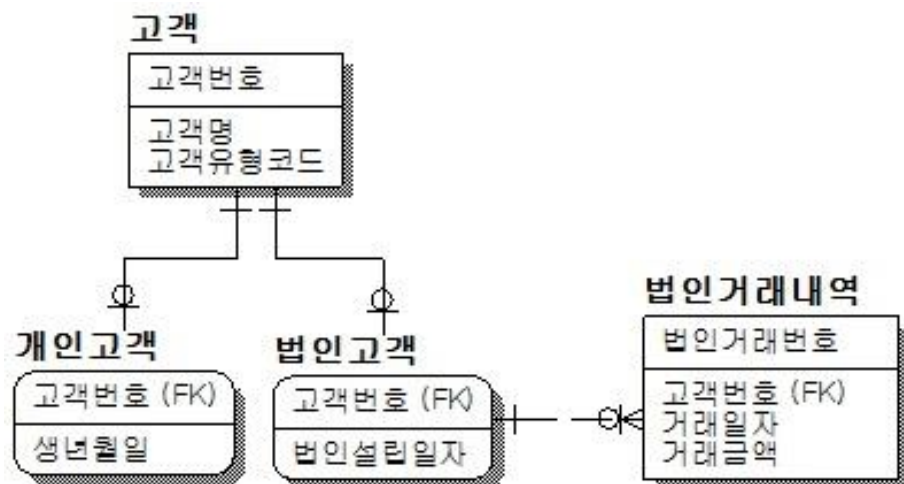
예) 하루에 여러 개 주문 가능하도록
주문일자 속성을 주문일시 속성으로 바꾸면
하위 엔티티에 미치는 영향도 매우 커지게 된다

예) 주문번호
인조식별자 사용: 하루에 여러 개 주문 가능

[예제] 참조무결성 제약조건



예) 문제점 발생
법인거래내역 엔티티



[고객]

#고객번호	고객명	고객유형코드
123	홍길동	개인
234	김길동	개인
890	(주)오성	법인
345	강길동	개인
789	(주)이성	법인

[개인고객]

#고객번호	생년월일
123	2030-01-06
234	2028-10-15
345	2036-07-07

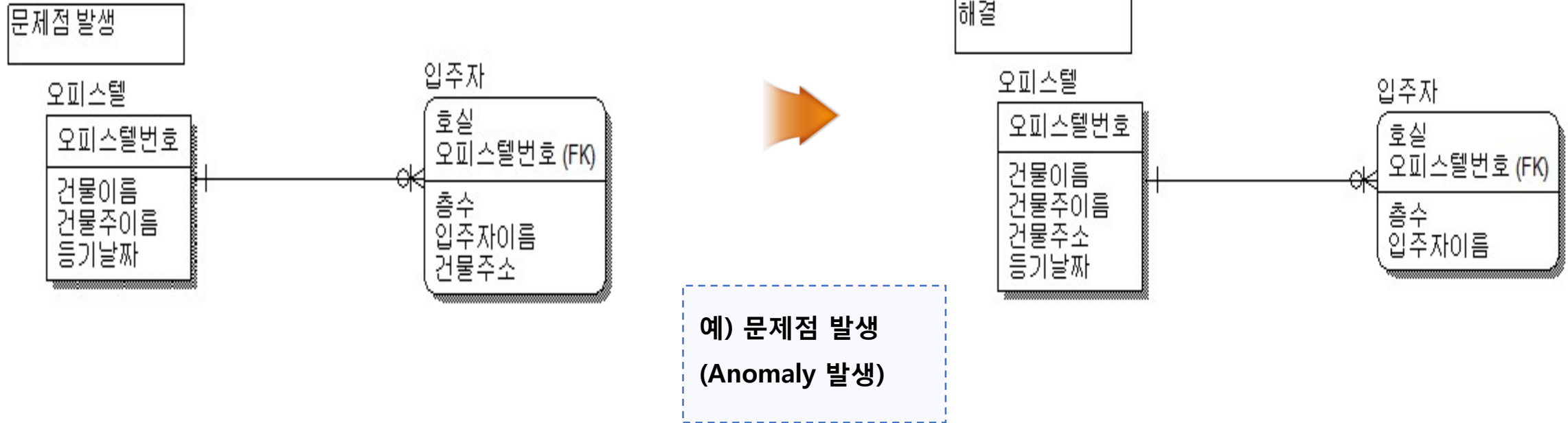
[법인고객]

#고객번호	법인설립일자
890	2021-01-06
789	2023-10-15

[법인거래내역]

#법인거래번호	거래일자	거래금액	고객번호
1000	2053-06-06	1000	890
1001	2053-06-07	1500	345
1002	2053-06-08	1000	789

[예제] 인스턴스 반복 출현 (Anomaly 발생)



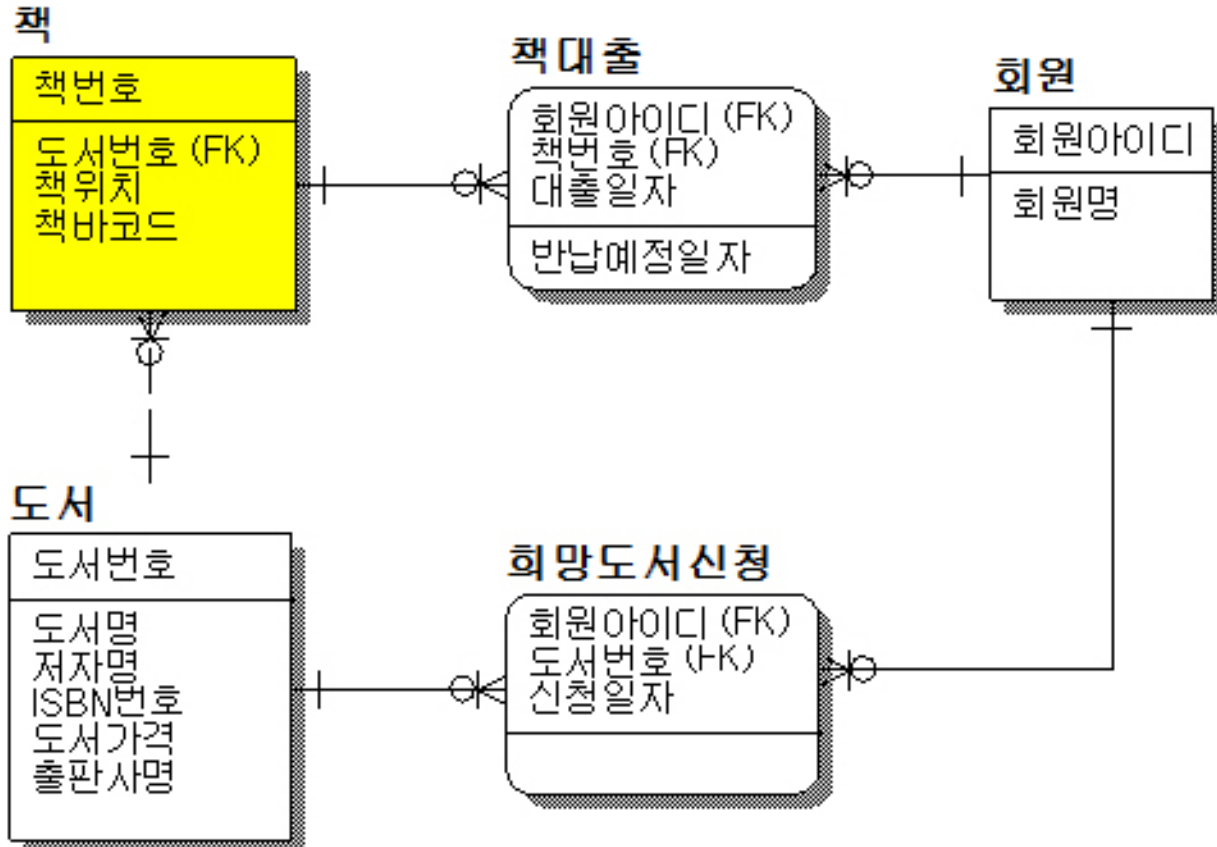
모델의 검토: 도서관

12주차 2-02

담당교수: 김희숙
(jasmin11@hanmail.net)

[예제] 실체 엔티티

❖ **실체 엔티티**: 실제로 보이는 것을 나타내는 데이터를 관리하는 엔티티



도서관 업무 설계 모델:

책을 대출할 수 있고 희망도서를 신청할 수 있는 간략한 모델

1) **도서관에 있는 책**: 실체 엔티티인가?

대출: 실제 책을 손에 들고 빌린다

2) **희망도서**: 실체 엔티티인가?

책을 나타내는 정보를 지목해서 구매요청

희망도서를 신청할 때는 실제 책이 아니라 책의 개념을 신청

1) **책**: 실체 엔티티

날권의 실제 책(인스턴스 책 개수만큼 여러 개 존재)

2) **도서**: 실체 엔티티 아니다

책 정보(인스턴스 한 개만 존재)

(예) 간단한 도서관 모델

[예제 12-06] 도서대출

대학의 도서관에서는 도서관 전산화를 위한 데이터베이스 구축을 위해 업무를 분석하고 있다.

도서관에 있는 책들은 단행본과 연속간행물로 구분되는데,
모든 책들은 도서번호, 도서명, 출판사, 구입일 등의 정보를 갖고 있다.

단행본의 경우에는 저자의 이름에 대한 정보가 있으며,
연속간행물의 경우에는 몇 권 몇 호인지에 대한 권과 호에 대한 정보가 있다.

대출장부에는 대출한 학생의 학번, 책의 도서번호, 대출일, 반납예정일을 기록해 둔다.

대출기간은 일주일이며, 연체할 경우 연체료를 부과한다.

또한 대출된 책은 대출예약을 신청할 수 있다.

대학 도서관에서는 책을 출판한 출판사에 대한 정보를 갖고 있는데,
여기에는 출판사 이름, 창립연도, 전화번호, 주소 등이 포함된다.

논문, 학술지, 단행본, 간행물

학위논문: 한 권의 논문(석박사 학위논문)

학술지: 전문학회에서 발간하는 논문집
(여러 편)

단행본: 책(일반 저서: 심사과정 불필요)

간행물: 잡지(일정 주기로 발간)

[예제 12-06] 도서대출

대학의 도서관에서는 도서관 전산화를 위한 데이터베이스 구축을 위해 업무를 분석하고 있다.

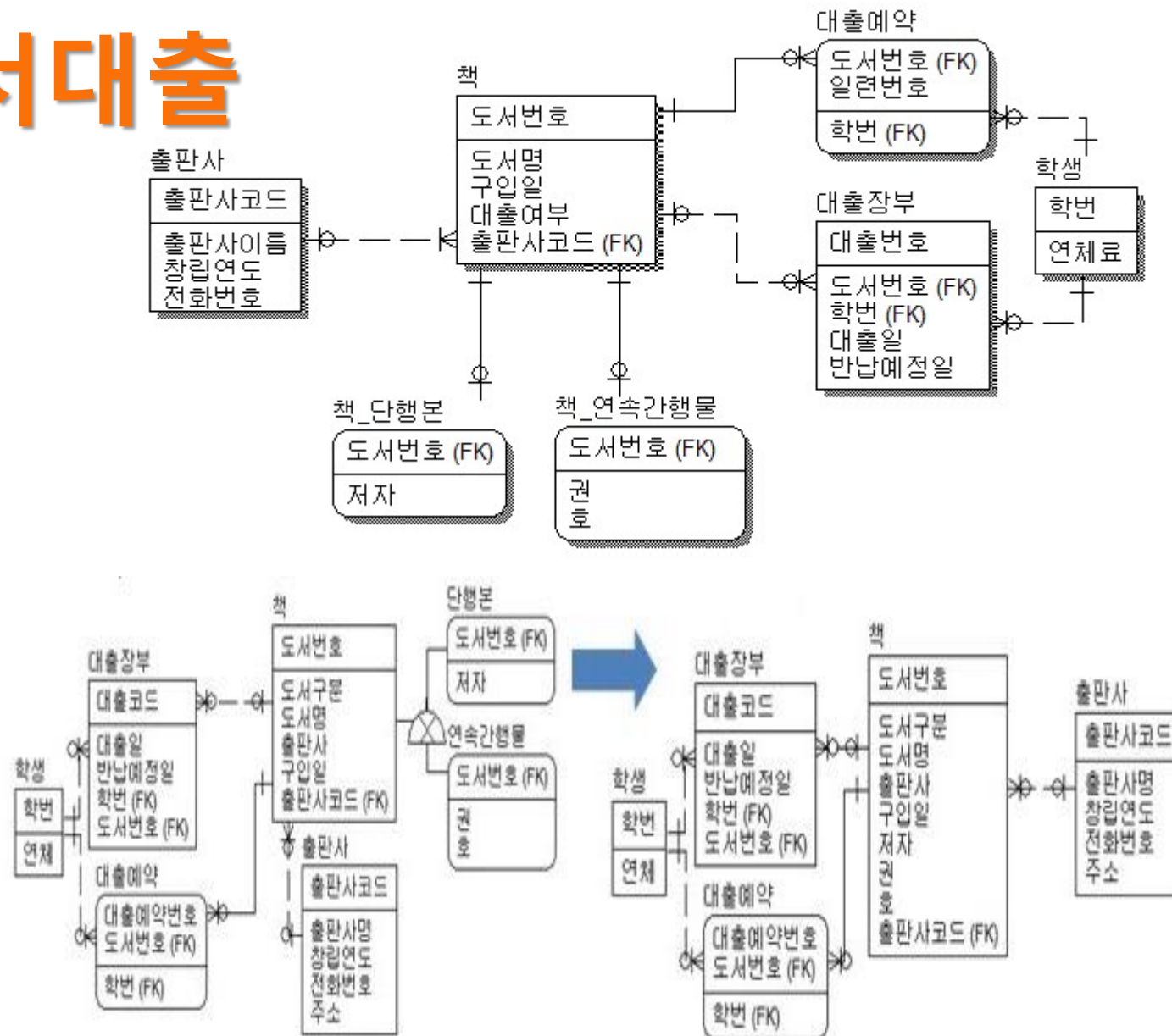
도서관에 있는 책들은 단행본과 연속간행물로 구분되는데, 모든 책들은 도서번호, 도서명, 출판사, 구입일 등의 정보를 갖고 있다. 단행본의 경우에는 저자의 이름에 대한 정보가 있으며, 연속간행물의 경우에는 몇 권 몇 호인지에 대한 권과 호에 대한 정보가 있다.

대출장부에는 대출한 학생의 학번, 책의 도서번호, 대출일, 반납예정일을 기록해 둔다.

대출기간은 일주일이며, 연체할 경우 연체료를 부과한다.

또한 대출된 책은 대출예약을 신청할 수 있다.

대학 도서관에서는 책을 출판한 출판사에 대한 정보를 갖고 있는데, 여기에는 출판사 이름, 창립연도, 전화번호, 주소 등이 포함된다.



[요약] 모델의 검토



1. 엔티티 검토

- 1) 엔티티가 적절히 구성되었는지 여부
- 2) 주식별자가 적절히 선정되었는지 여부

2. 속성 검토

- 1) 여러 엔티티 사이에 중복된 속성이 존재하지는 않는가
- 2) 낱짜를 저장하는 속성이 올바르게 구성되었는가
- 3) 속성의 성격상 코드화해야 하는 것은 없는가

3. 관계 검토

- 1) ERD상에서 다른 엔티티와 관계없이 독립적으로 존재하는 엔티티는 없는가
- 2) 관계를 너무 복잡하게 맺지는 않았는가



10.2 엔티티의 검토

- 각 엔티티는 현실세계의 정보를 효과적으로 관리할 수 있는 구조인가

도서
◆ 도서관리번호
분류기호
제목
저자
출판사
출판연도
구입일자
국내/해외구분
신간도서여부
기타정보
세금계산서번호(FK)

인기 있는 작품은 같은 책을 여러 권 비치할 수 있다

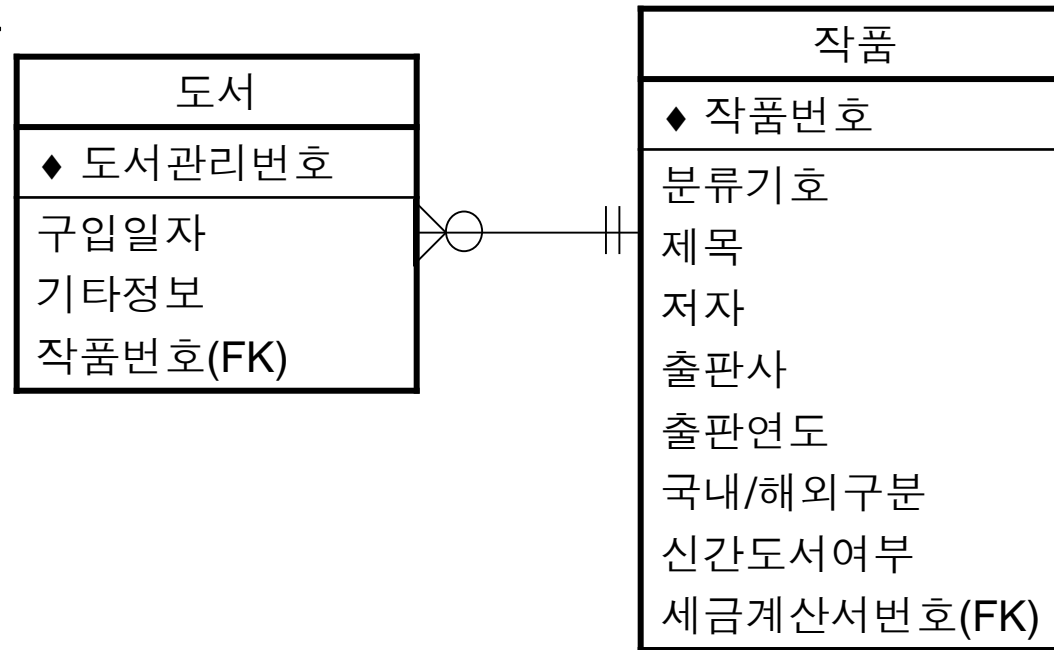
10.2 엔티티의 검토

- 각 엔티티는 현실세계의 정보를 효과적으로 관리할 수 있는 구조인가

관리 번호	제목	저자	출판사	국내해외 구분	기타 정보	신간도 서여부
1101	The Scarlet Letter	N. Hathorne	Heaven	O		N
1102	조선왕조실록	오병욱	하나출판	D		Y
1103	조선왕조실록	오병욱	하나출판	D		Y
1104	조선왕조실록	오병욱	하나출판	D		Y
1105	조선왕조실록	오병욱	하나출판	D		N
1106	The good earch	P. Buck	Andrew	O		N

동일
작품

<개선된 모델>



관리 번호	기타	작품 번호
1101	-	A001
1102	-	A002
1103	-	A002
1104	-	A002
1105	-	A002
1106	파손	A003

작품 번호	제목	저자	출판사	국내해 외구분	신간도 서여부
A001	The Scarlet Letter	N. Hathorne	Heaven	O	N
A002	조선왕조실록	오병욱	하나출판	D	Y
A003	The good earch	P. Buck	Andrew	O	N

10.2 엔티티의 검토

- 각 엔티티는 현실세계의 정보를 효과적으로 관리할 수 있는 구조인가

대출
◆ 대출번호
회원번호(FK)
도서관리번호(FK)
대출일
반납예정일
반납여부
반납구분
연체료납부여부

한번에 여러 권의
도서를 대출할 수 있다

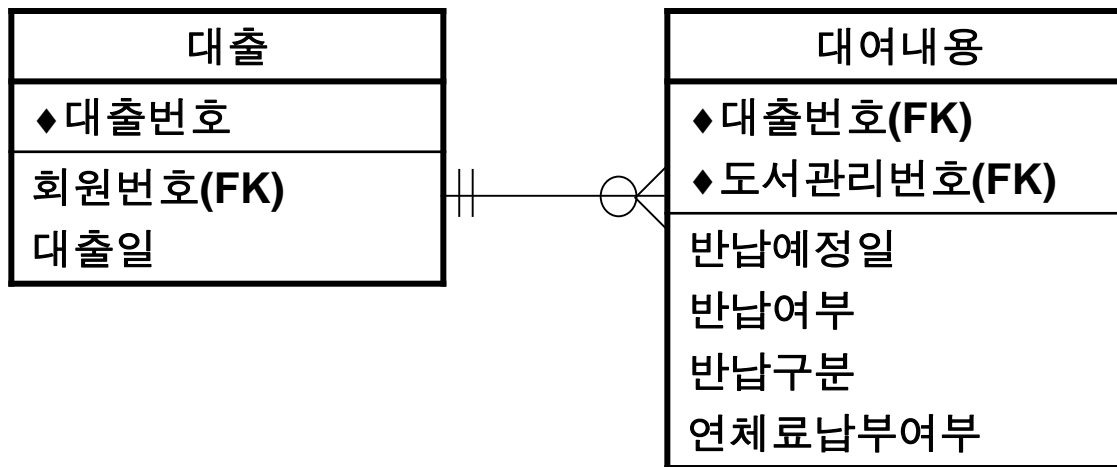
10.2 엔티티의 검토

- 각 엔티티는 현실세계의 정보를 효과적으로 관리할 수 있는 구조인가

대출 번호	회원번호	도서관리 번호	대출일	반납예정일	반납여부	연체료 납부여부
91001	101	1109	2011.11.1	2005.11.3	Y	N
91002	103	1101	2011.11.9	2005.11.10	N	N
91003	103	1105	2011.11.9	2005.11.11.	N	N
91004	103	1210	2011.11.9	2005.11.10	N	N

} 대여
1회

<개선된 모델>

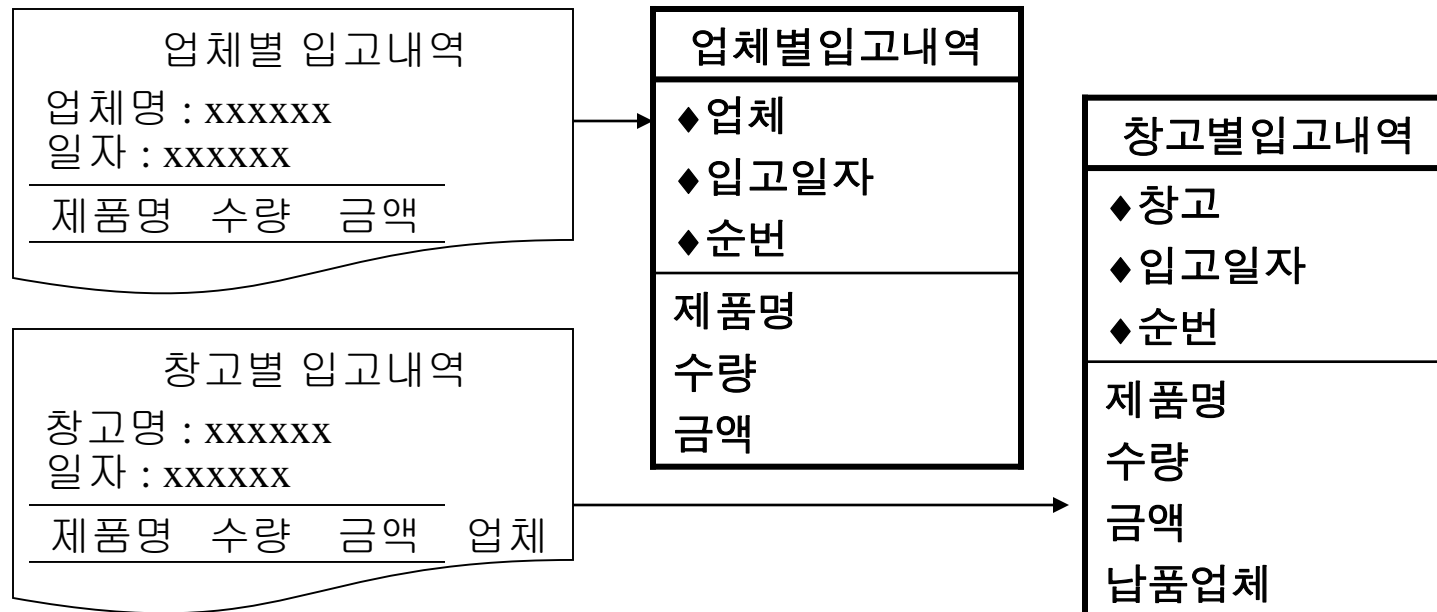


대출 번호	회원번호	대출일
91001	101	2011.11.1
91002	103	2011.11.9

대출 번호	도서관리 번호	반납예정일	반납 여부	연체료 납부여부
91001	1109	2005.11.3	Y	N
91002	1101	2005.11.10	N	N
91002	1105	2005.11.11.	N	N
91002	1210	2005.11.10	N	N

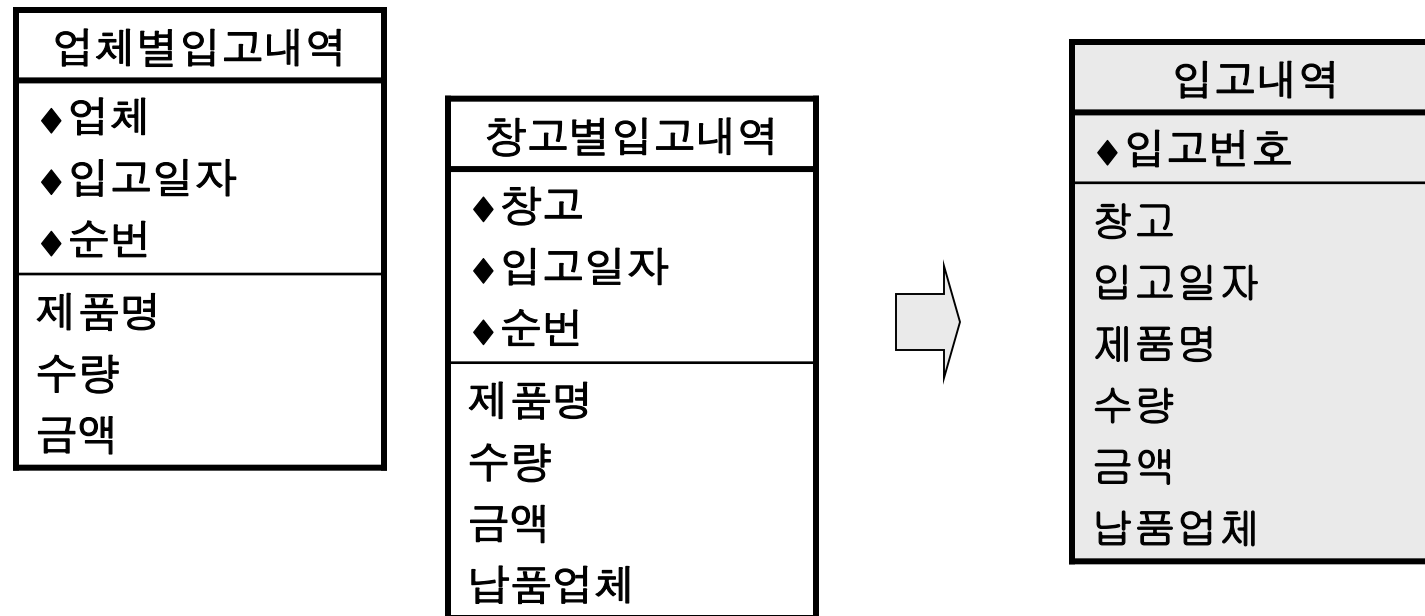
10.2 엔티티의 검토

- 유사한 내용을 관리하는 엔티티는 없는가
 - 여러 설계자가 동일 내용을 서로 다른 엔티티로 표현
 - 한 설계자가 장부나 전표, 레포트를 각각 하나의 엔티티로 표현



10.2 엔티티의 검토

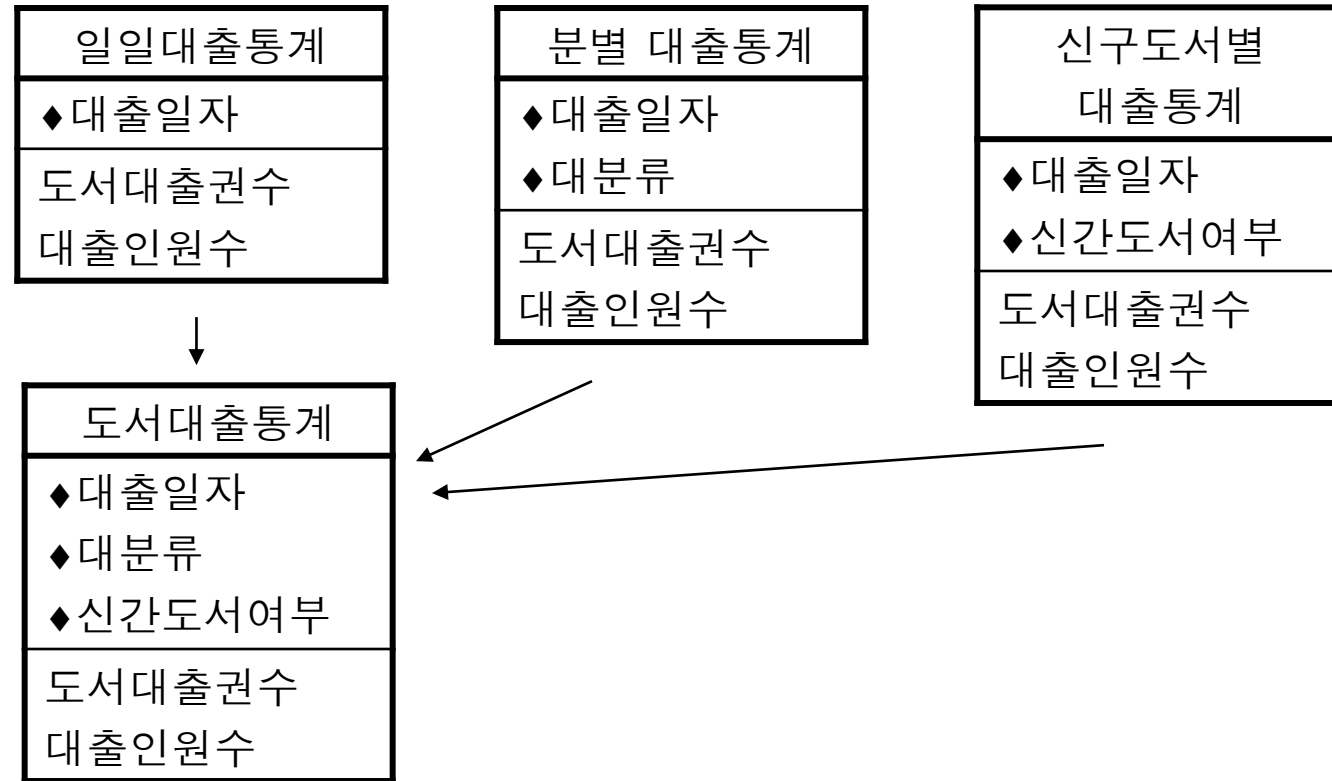
- 유사한 내용을 관리하는 엔티티는 없는가



<그림 10.8> 관리내용이 유사한 엔티티의 통합

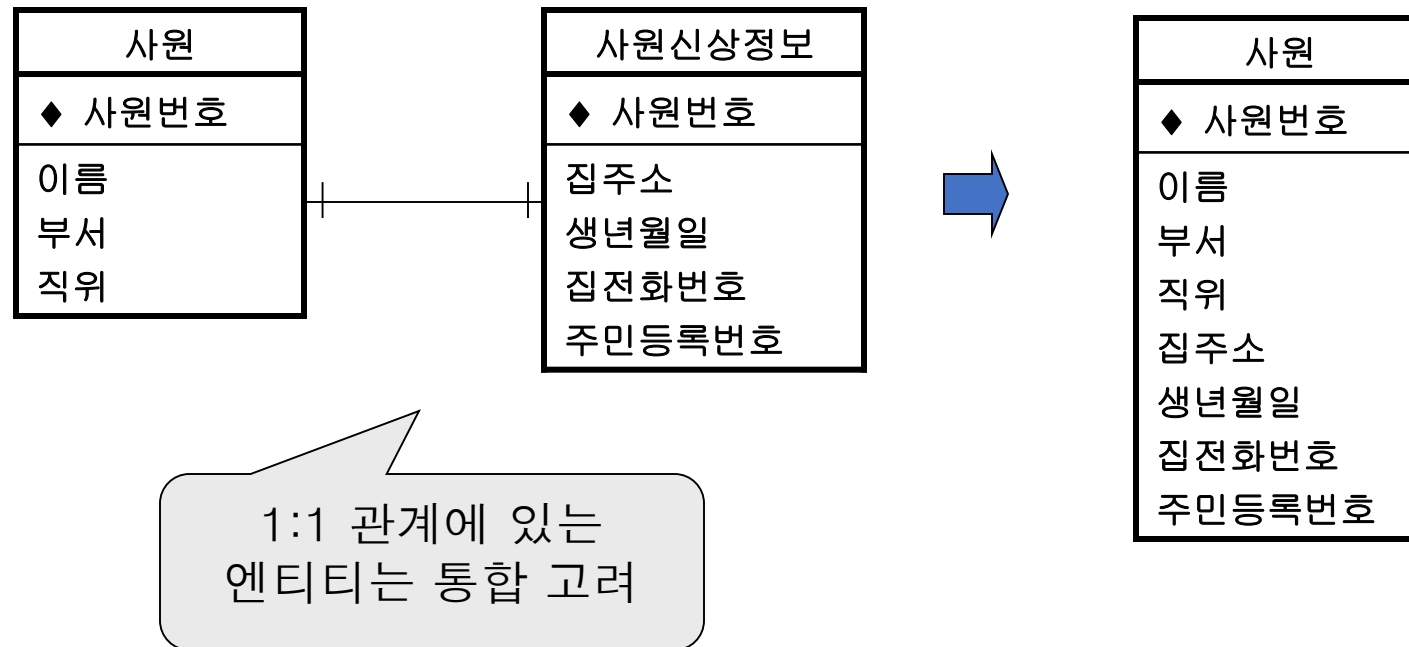
10.2 엔티티의 검토

- 유사한 내용을 관리하는 엔티티는 없는가
 - 여러 종류의 통계 엔티티 통합



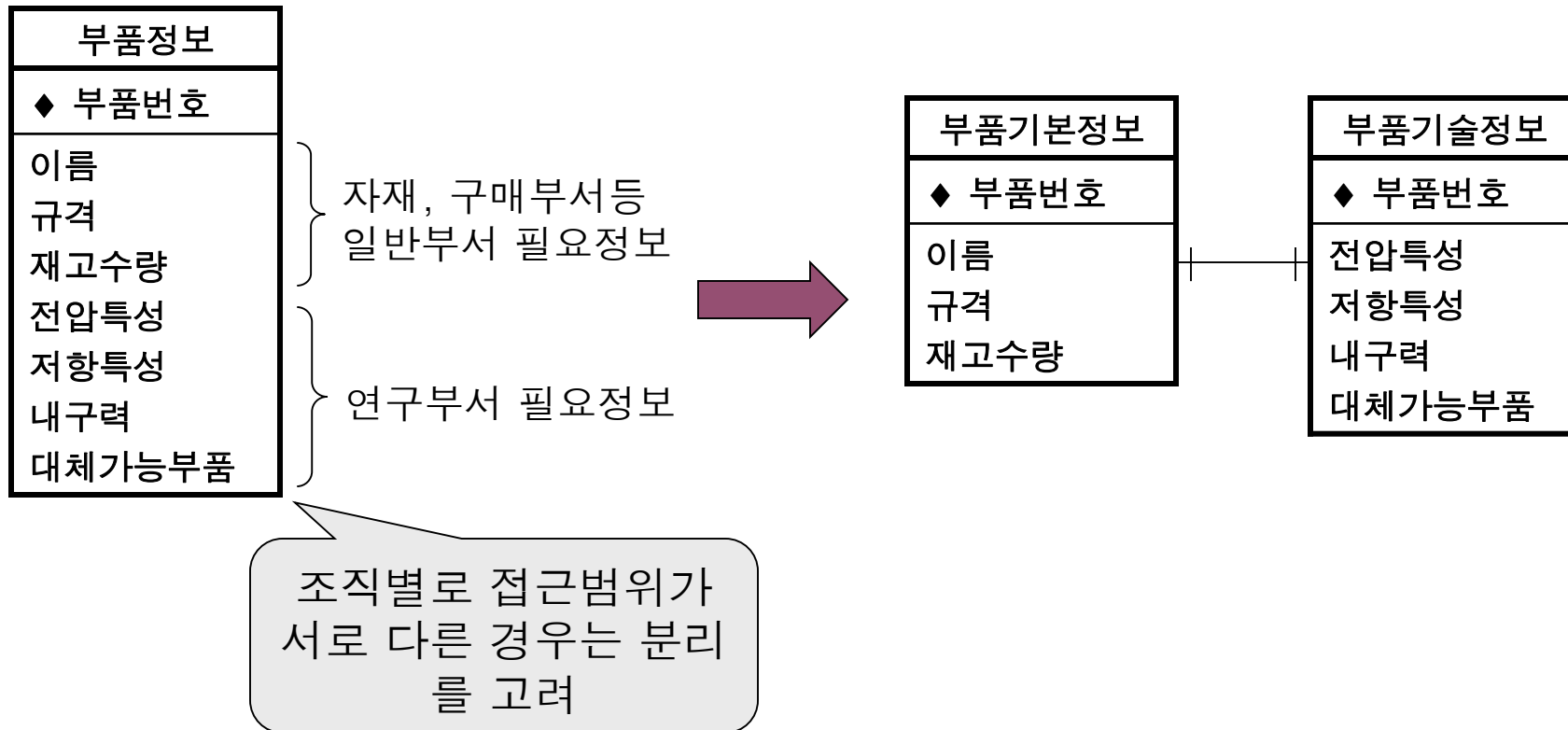
10.2 엔티티의 검토

- 통합 또는 분리되어야 할 엔티티는 없는가



10.2 엔티티의 검토

- 통합 또는 분리되어야 할 엔티티는 없는가



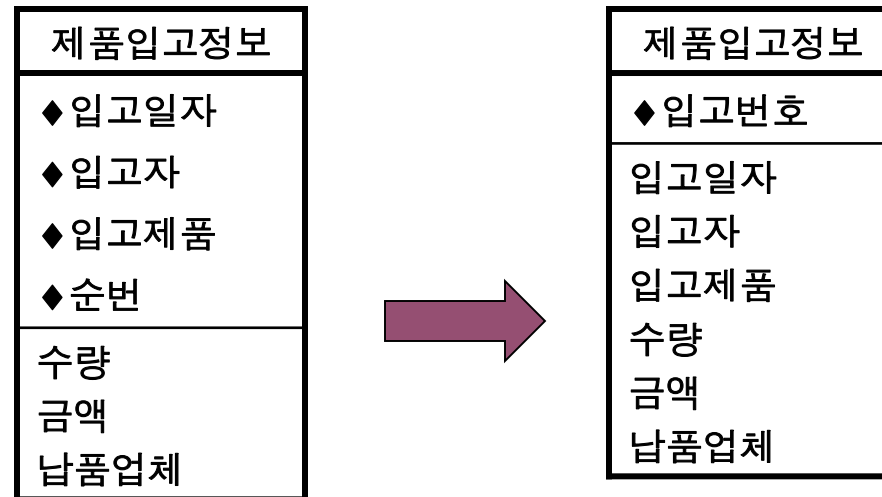
10.2 엔티티의 검토



- 주식별자는 인스턴스의 유일성을 보장해 주는가
 - 주식별자 원칙에 따라 올바르게 선정되었는가
- 주식별자에 불필요한 속성이 포함되어 있지는 않은가
 - 주식별자는 최소한의 속성 혹은 속성들로 구성되었는가

10.2 엔티티의 검토

- 주식별자에 속성이 너무 많지는 않은가



<그림 10.14> 복잡한 주식별자의 단순화

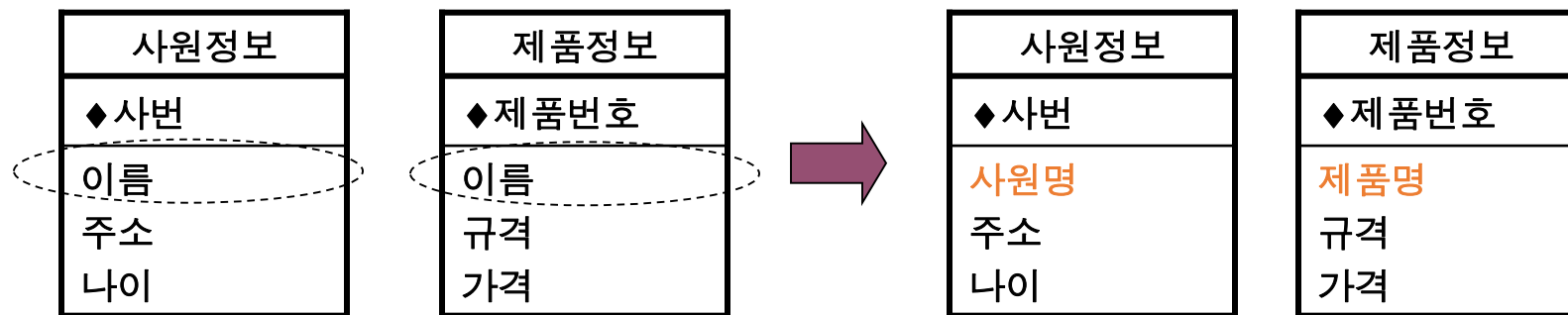
10.3 속성의 검토

- 검토사항
 - 여러 엔티티 사이에 중복된 속성이 존재하지는 않는가?
 - 날짜를 저장하는 속성이 올바르게 구성되었는가?
 - 속성의 성격상 코드화 해야하는 것은 없는가?



10.3 속성의 검토

- 여러 엔티티 사이에 중복된 속성이 존재하지는 않는가?
 - 같은 속성 이름은 전 ERD에서 한번만 나타나는 것이 원칙
 - 서로 다른 내용이 같은 속성 이름으로 되어 있다면 이름을 다르게 한다

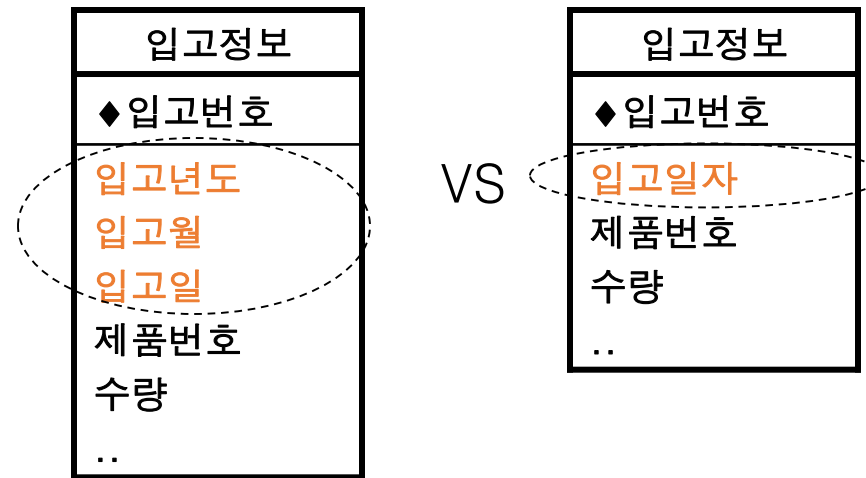


<그림 10.15> 속성 이름의 변경이 바람직한 경우

10.3 속성의 검토

- 날짜를 저장하는 속성이 올바르게 구성되었는가?

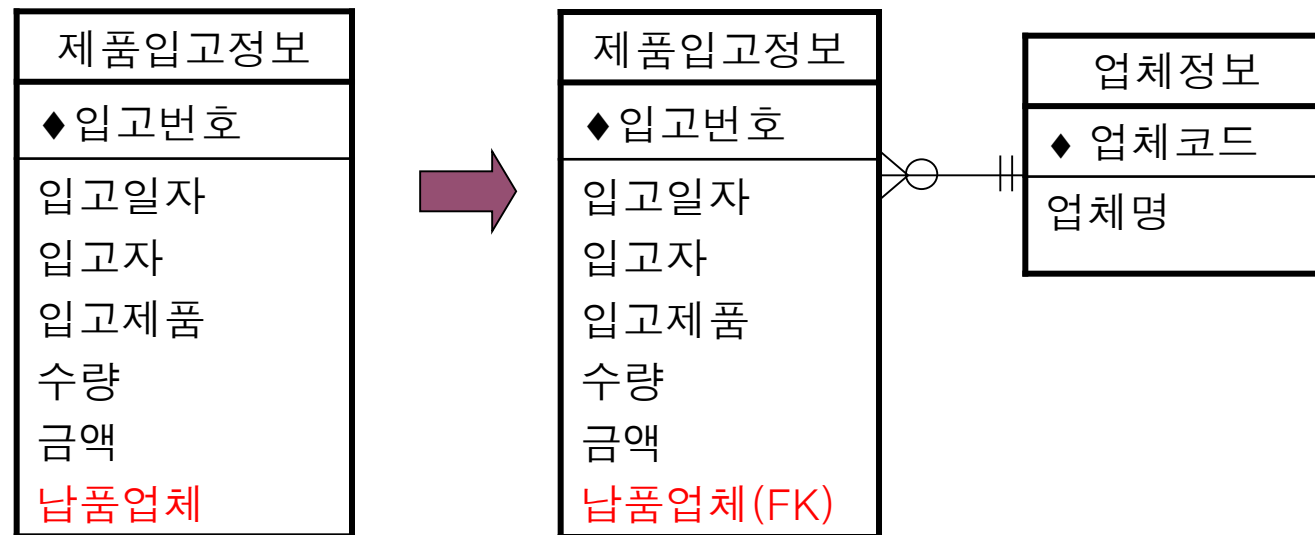
통계를 위한 엔티티의 경우는 연도별, 월별, 일별로 데이터를 저장하는 것이 관리를 위해 필요한 경우도 있다



<그림 10.16> 날짜 속성의 표현방법

10.3 속성의 검토

- 속성의 성격상 코드화 해야하는 것은 없는가?
 - 속성들 중에는 값을 직접 저장하는 것 보다는 코드 엔티티를 만들고 코드 값을 저장하는 것이 바람직한 경우가 있다



<그림 10.17> 속성의 코드화

10.3 속성의 검토

- 모델의 검토 단계에서 속성의 추가 사례

이용자에게 발송할 우편물에
붙일 주소 라벨을 출력할 수 있
도록 한다.

회원
◆ 회원번호
회원이름
전화번호
집주소
우편번호
이메일
탈퇴여부
우수회원 여 부
미반납여부
가입일자

추가

10.3 속성의 검토



- 모델의 검토 단계에서 속성의 추가 사례

미납자 현황을 원하는 시점에서 즉시 확인할 수 있어야 한다.

→ 추가할 사항 없음

월간 도서 대출 순위를 작성할 수 있어야 한다..

→ 추가할 사항 없음

10.3 속성의 검토

- 모델의 검토 단계에서 속성의 추가 사례
 - 시스템의 운영을 위해 여러 환경변수를 저장할 필요가 있다.

환경변수
◆ 환경변수명
환경변수 값
환경변수 설명

← 엔티티의
추가

환경변수명	환경변수 값	환경변수 설명
LENT_BSET	10	대출 순위를 몇위까지 계산하는가
ADMIN_ID	mario	관리자의 ID 명
POPUP_YN	N	프로그램 시작시 pop up 광고를 보일지 여부

10.3 속성의 검토

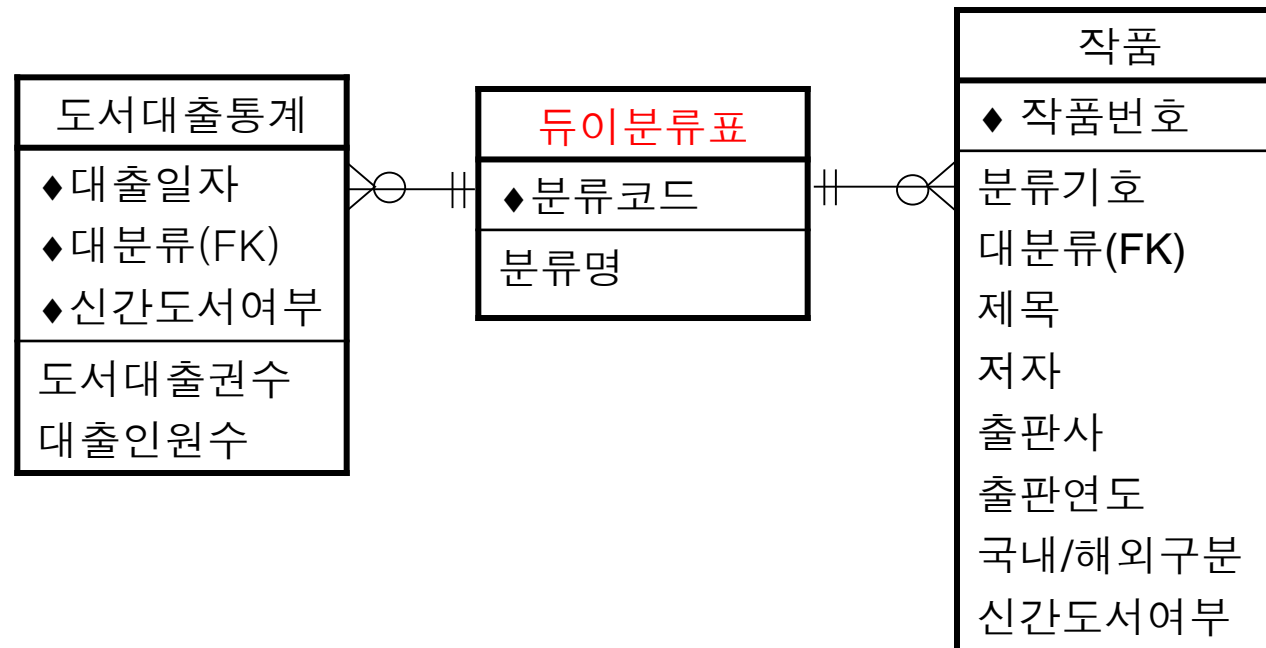
- 도서 분류:
듀이 분류표를 따르므로 이에 관련된 엔티티, 속성 추가

000	총류
100	철학
200	종교
300	사회과학
400	언어
500	자연과학
600	응용과학
700	예술
800	문학
900	역사및지리

<듀이 십진분류표>

10.3 속성의 검토

- 듀이분류표 엔티티의 추가



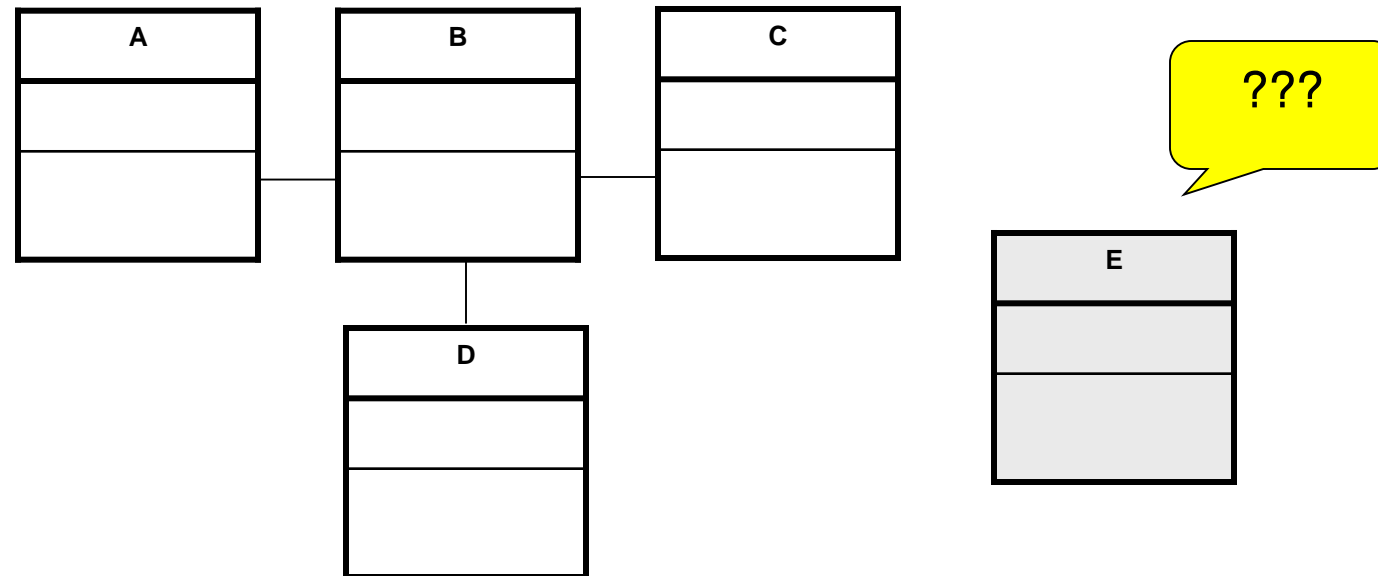
10.4 관계의 검토

- 검토사항
 - ERD상에서 다른 엔티티와 관계가 없이 독립적으로 존재하는 엔티티는 없는가
 - 관계를 너무 복잡하게 맺지는 않았는가



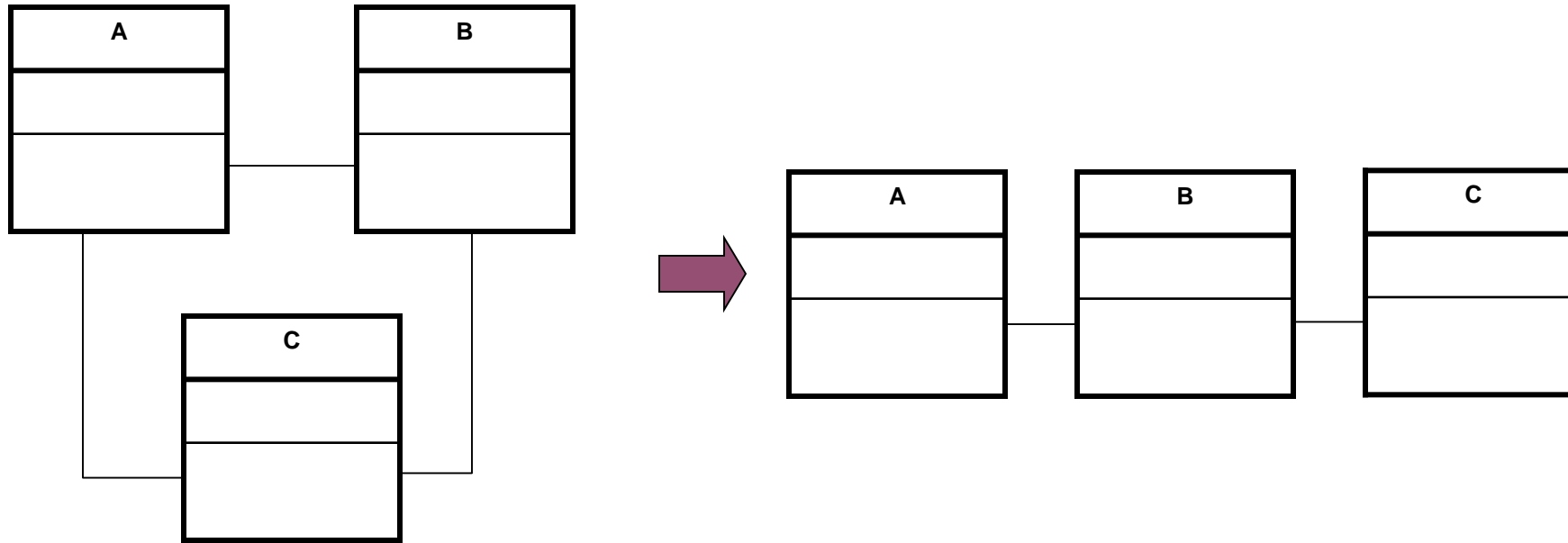
10.4 관계의 검토

- ERD상에서 다른 엔티티와 관계가 없이 독립적으로 존재하는 엔티티는 없는가
 - 엔티티들은 ERD상에서 상호 관계를 가져야한다 (통계 엔티티, 환경변수 엔티티를 제외)



10.4 관계의 검토

- 관계를 너무 복잡하게 맺지는 않았는가



<그림 10.19> 복잡한 관계의 단순화

[예제] 도서관관리

