

- 식별과 비식별을 선택하여 아래의 예제의 관계 설정을 해주세요.

지역

지역번호
지역명

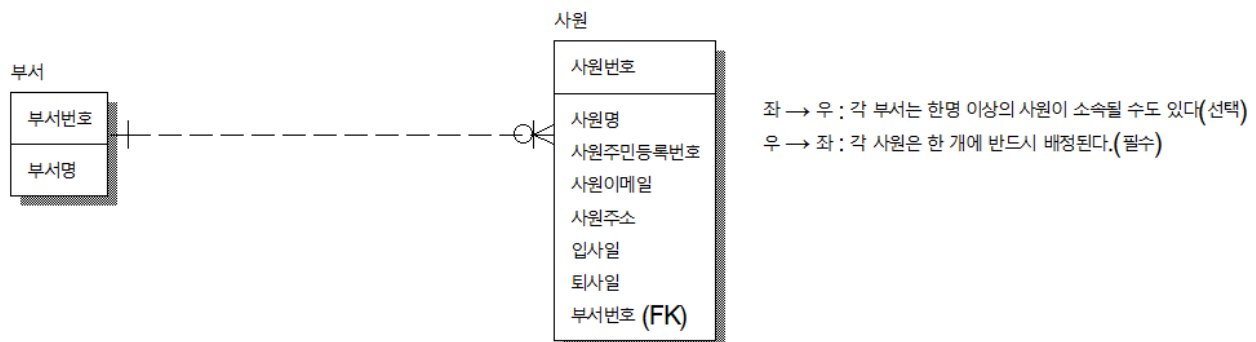
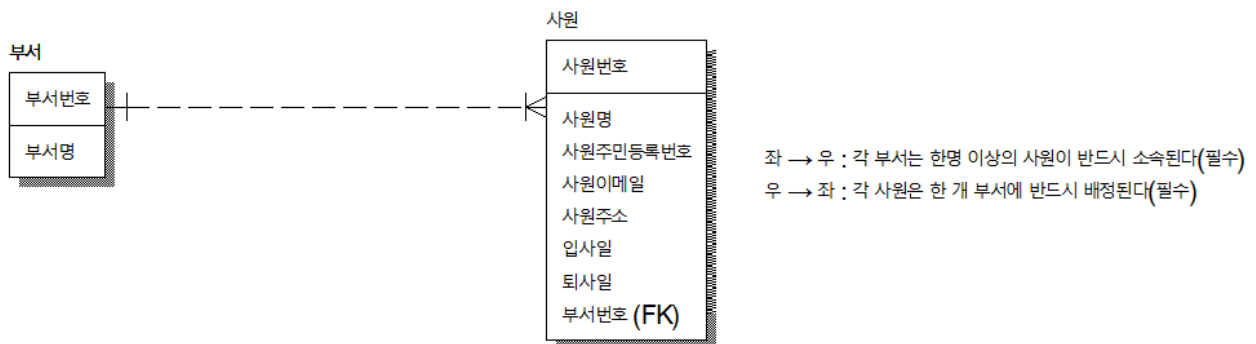
부서

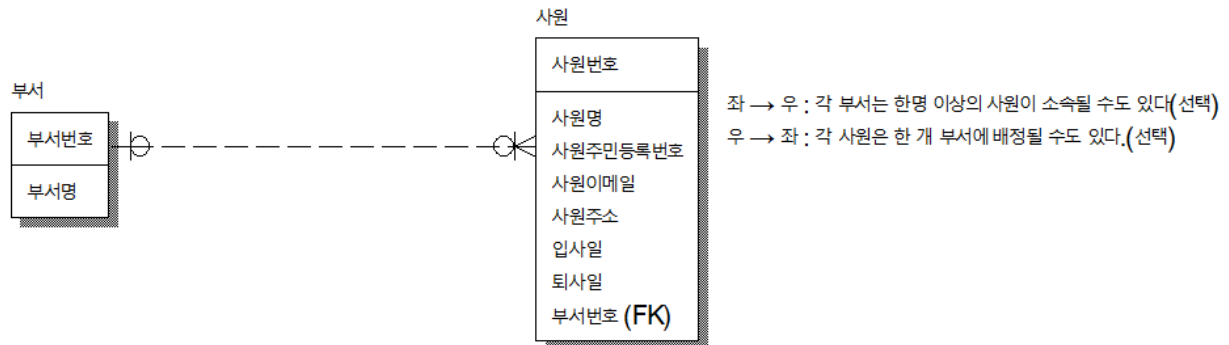
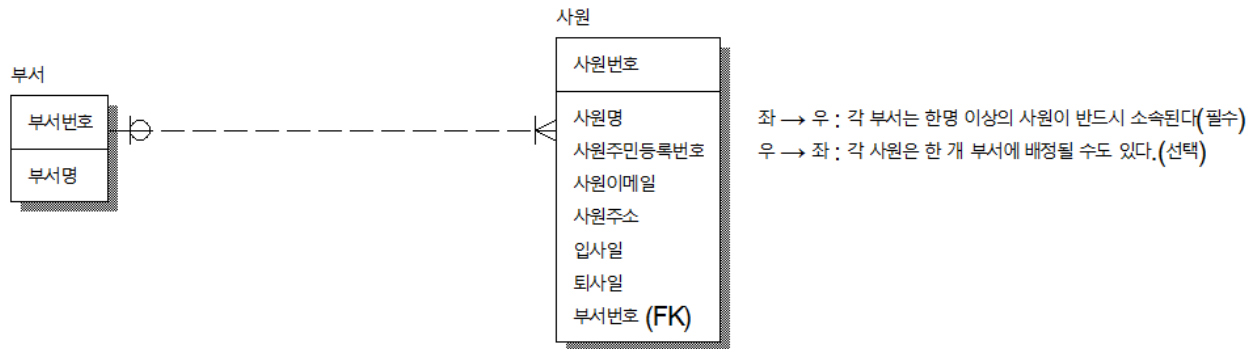
부서번호
부서명

직원

직원번호
직원명
직원주민등록번호
직원이메일
직원주소
입사일
퇴사일

기호	의미
-----	<ul style="list-style-type: none"> <li>• 비식별자 관계(non-identifying relationship): 강한 개체 타입</li> <li>• 부모 개체의 키가 일반 속성으로 포함되는 관계</li> </ul>
_____	<ul style="list-style-type: none"> <li>• 식별자 관계(identifying relationship): 약한 개체 타입</li> <li>• 부모 개체의 키가 주식별자로 포함되는 관계</li> </ul>
----->	<ul style="list-style-type: none"> <li>• 일대다(1:N)의 관계: N 쪽에 새발을 표시</li> </ul>
-----○	<ul style="list-style-type: none"> <li>• 0(선택 참여, 최소 참여가 0일 경우)</li> </ul>
-----+	<ul style="list-style-type: none"> <li>• 1(필수 참여, 최소 참여가 1일 경우)</li> </ul>









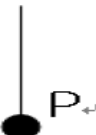



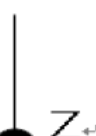
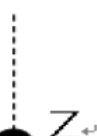




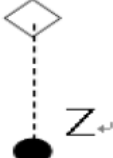

## 1. 관계 정의 설정

공백 : Parent와 Child 사이에 0, 1, 다수의 관계

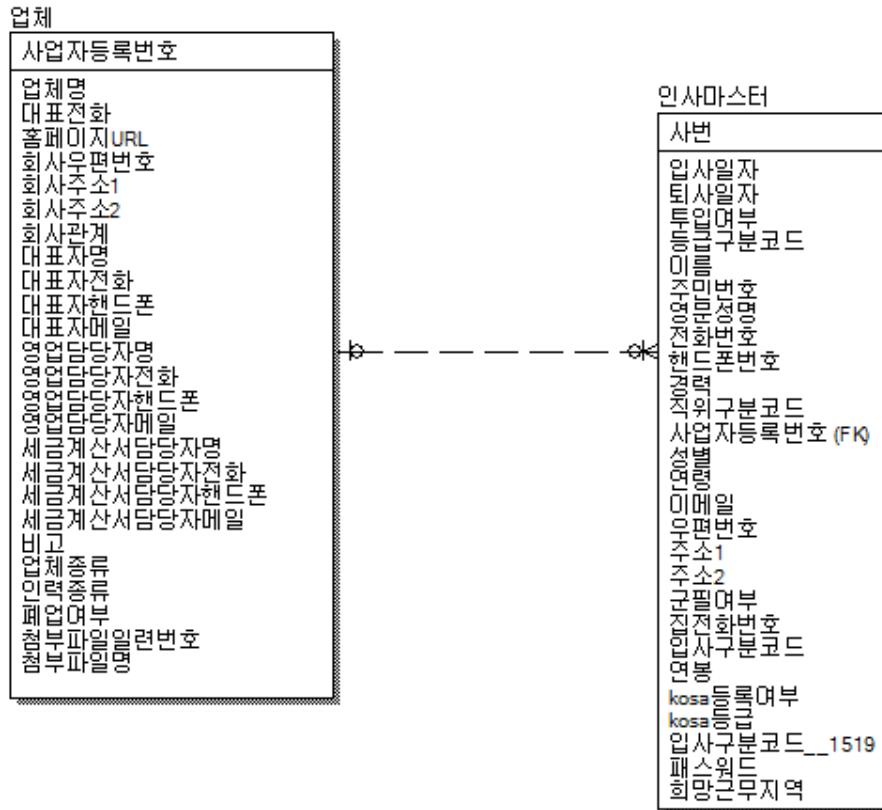
P : Parent와 Child 사이에 1, 다수의 관계

Z : Parent와 Child 사이에 0, 1의 관계

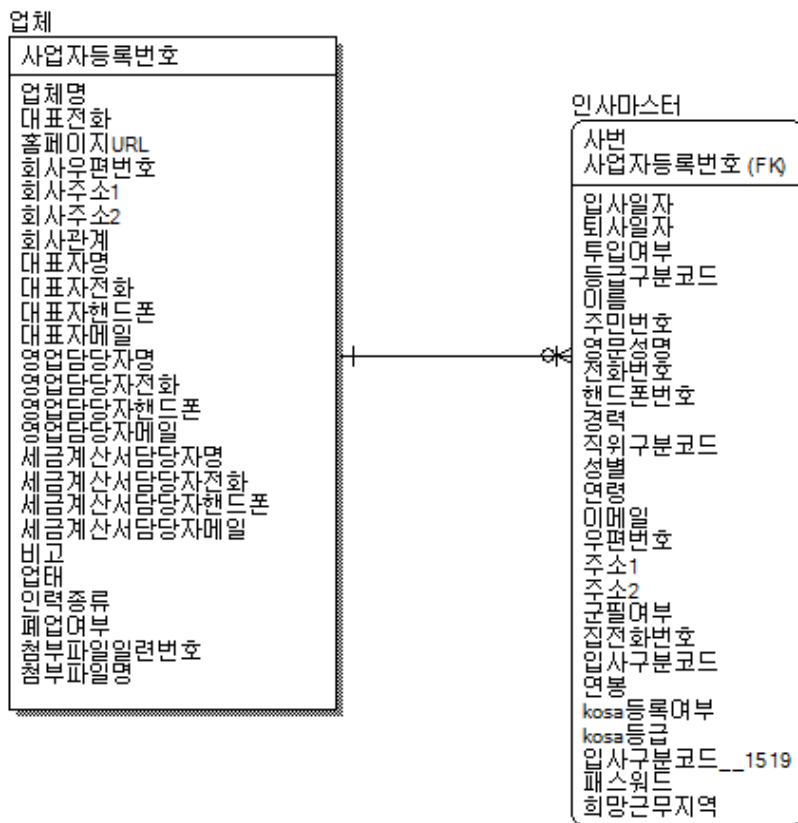
Exactly "n" : Parent와 Child 사이에 정확히 N개인 관계

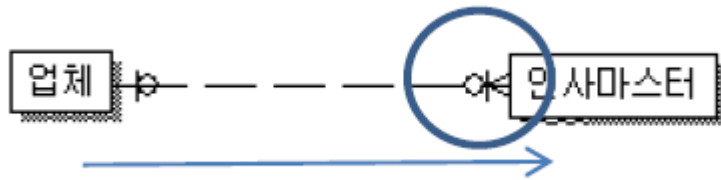
Cardinality Description	IDEFIX Identifying Non-identifying	IE Identifying Non-identifying
One to Zero, one or more	 	 
One to one or more	 	 
One to zero or one	 	 
Zero or one to zero, one, or none(non-identifying only)		
Zero or one to zero or one(non-identifying only)		

둘의 관계는 비식별(Non-identifying)자의 관계로 연결되어 있다. 정의해보면 "인사마스터는 소속된 업체가 하나 있을 수도 있고 없을 수도 있다"라고 할 수 있다.



만약에 둘의 관계가 식별이었다면 "인사마스터는 항상 소속된 업체를 가진다"라고 정의할 수 있다.





관련기호를 살펴보면

- | 꼭 존재해야 하는(mandatory)
- O 존재해도 되는(optional) - 존재할 수도 존재 안 할수도 있는
- < 여러 개 존재할 수 있는

그래서 업체에서 인사마스터를 정의해 보면 "업체는 인사마스터를 안 가질 수도 있고 한 개 가질 수도 있고 여러 개 가질 수도 있다" 라고 정의할 수 있다.



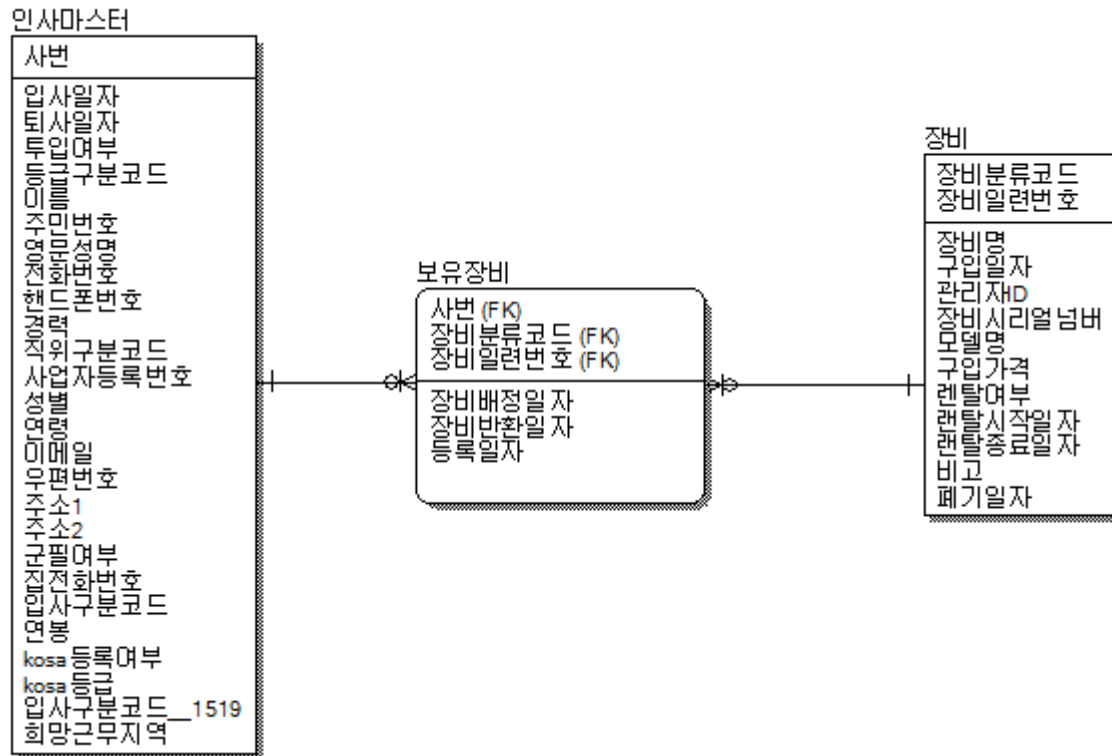
인사마스터에서 업체를 정의해보면

인사마스터는 업체를 가질 수도 있고 한 개 가질 수도 있다.



정의해 보면 사원은 여러 개의 장비를 가지고 있다라고 정의되고 장비는 여러 명의 사원에 지급될 수

있다고 정의할 수 있다. 이때는 관계를 해소해주는 교차 엔티티를 만들어 N:M의 관계를 해소해 준다



보유장비 엔티티를 보면 사원에 대한 장비를 찾아 낼 수 있고, 지급된 장비에 대한 여러 사원을 찾을 수 있는 연결고리가 성립되어 있다. 인사마스터 : 보유장비는 1:M, 보유장비 : 장비는 M:1 관계로 풀려 있다.

## 정규화

정규화는 관계형 데이터베이스를 모델링할 때 거쳐야 하는 필수 작업이다. 정규화란 데이터 모델을 보다 효율적으로 개선시켜나가는 과정을 뜻한다. 즉 고유한 식별자(Primary Key)를 가지는 개체(테이블)에 대하여 더 이상 분리할 수 없는 상태로 나누는 과정을 말한다.

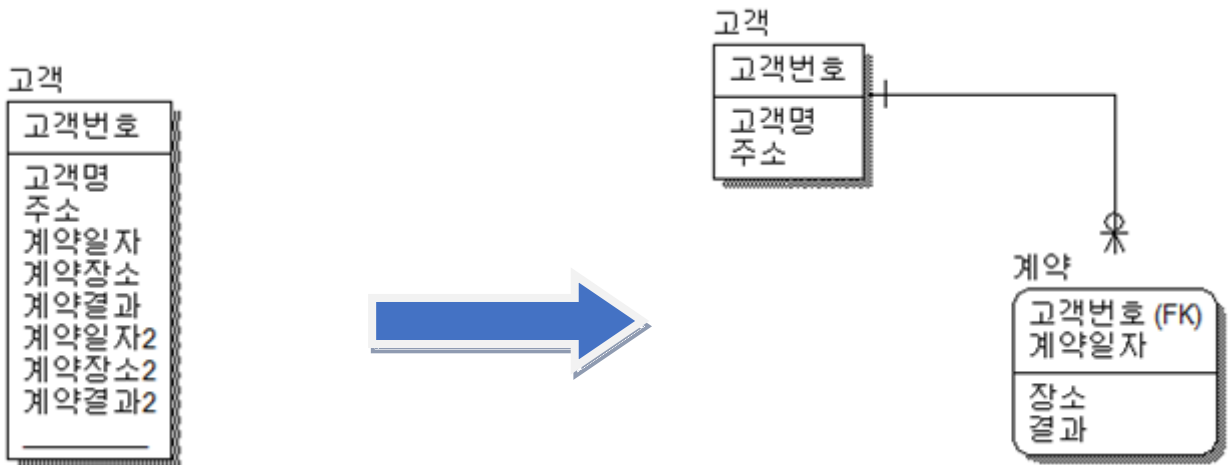
일반적으로 정규화(normalization)는 **데이터베이스에 저장되는 데이터 중복의 최소화**하기 위하여 서로 독립적인 관계는 별개의 테이블로 표현한다는 것이다.

### 제1정규화

반복되는 속성들을 다른 개체(테이블)로 나누어 분리하는 작업이다.

또 다른 개체로 판단할 수 있는 속성들을 분리하고, 각 객체 속성들의 유일한 식별자를 갖게된다. 부모 테이블의 식별자는 자식 테이블의 외부키와 연결된다.

- 모든 속성은 반드시 하나의 값을 가져야 한다. (반복형태가 있어서는 안된다.)
- 정규화가 이루어지고 나면 자식 엔티티로 분리된다.



정규화 되기 전 데이터 구조

고객번호	고객명	주소	계약일자	계약장소	계약결과	계약일자2	계약장소2	계약결과2
2012010101	홍길동1	서울시 양천구 목4동 123-1 대성빌라 301호	20120101	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함	20120111	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010102	홍길동2	서울시 양천구 목4동 123-1 대성빌라 301호	20120102	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함	20120112	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010103	홍길동3	서울시 양천구 목4동 123-1 대성빌라 301호	20120103	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함	20120113	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010104	홍길동4	서울시 양천구 목4동 123-1 대성빌라 301호	20120104	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함	20120114	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010105	홍길동5	서울시 양천구 목4동 123-1 대성빌라 301호	20120105	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함			
2012010106	홍길동6	서울시 양천구 목4동 123-1 대성빌라 301호	20120106	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함			
2012010107	홍길동7	서울시 양천구 목4동 123-1 대성빌라 301호	20120107	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함			
2012010108	홍길동8	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010109	홍길동9	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010110	홍길동10	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010111	홍길동11	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010112	홍길동12	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010113	홍길동13	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010114	홍길동14	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010115	홍길동15	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010116	홍길동16	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010117	홍길동17	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010118	홍길동18	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010119	홍길동19	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010120	홍길동20	서울시 양천구 목4동 123-1 대성빌라 301호						

## 정규화된 후의 데이터 구조

고객 테이블

계약 테이블

고객번호	고객명	주소			고객번호	계약일자	계약장소	계약결과
2012010101	홍길동1	서울시 양천구 목4동 123-1 대성빌라 301호			2012010101	20120101	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010102	홍길동2	서울시 양천구 목4동 123-1 대성빌라 301호			2012010101	20120111	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010103	홍길동3	서울시 양천구 목4동 123-1 대성빌라 301호			2012010102	20120102	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010104	홍길동4	서울시 양천구 목4동 123-1 대성빌라 301호			2012010102	20120112	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010105	홍길동5	서울시 양천구 목4동 123-1 대성빌라 301호			2012010103	20120103	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010106	홍길동6	서울시 양천구 목4동 123-1 대성빌라 301호			2012010103	20120113	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010107	홍길동7	서울시 양천구 목4동 123-1 대성빌라 301호			2012010104	20120104	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010108	홍길동8	서울시 양천구 목4동 123-1 대성빌라 301호			2012010104	20120114	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010109	홍길동9	서울시 양천구 목4동 123-1 대성빌라 301호			2012010105	20120105	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010110	홍길동10	서울시 양천구 목4동 123-1 대성빌라 301호			2012010106	20120106	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010111	홍길동11	서울시 양천구 목4동 123-1 대성빌라 301호			2012010107	20120107	서울시 양천구 목4동 123-1 대성빌라 301호	조건부 계약을 함
2012010112	홍길동12	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010113	홍길동13	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010114	홍길동14	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010115	홍길동15	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010116	홍길동16	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010117	홍길동17	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010118	홍길동18	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010119	홍길동19	서울시 양천구 목4동 123-1 대성빌라 301호						
2012010120	홍길동20	서울시 양천구 목4동 123-1 대성빌라 301호						

## 제 1 정규형 제약조건

- 테이블의 모든 셀이 단일 값이어야 한다.

등록번호	등록일	학생번호	학생이름	과목번호	학점	성적
1	2006.2.20	1	김태희	1 2	2 2	90 83
2	2006.2.21	2	한채영	1	2	95
3	2006.2.22	1	김태희	3	3	75



제1정규형이 되기 위해서는 한 셀에 들어가 있는 여러 값을 분리하여 한 셀에 하나의 데이터만 나타나게 한다.

등록번호	등록일	학생번호	학생이름	과목번호	학점	성적
1	2006.2.20	1	김태희	1	2	90
1	2006.2.20	1	김태희	2	2	83
2	2006.2.21	2	한채영	1	2	95
3	2006.2.22	1	김태희	3	3	75

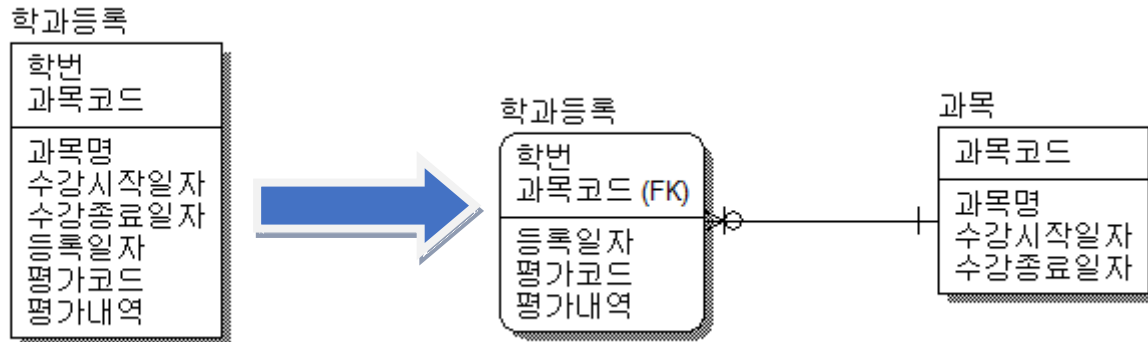


## 제 2정규화

모든 속성은 식별자에 직접적으로 의존적이어야 하며 이에 해당되지 않는 속성을 분리한다

모든 속성은 반드시 UID(Unique Identifier) 전부에 종속되어야 한다. (UID 일부에만 종속되어서는 안된다)

- 정규화가 이루어지고 나면 부모 엔티티로 분리된다.



## 정규화되기 전의 데이터구조

학과등록 테이블

학번	과목코드	과목명	수강시작일자	수강종료일자	등록일자
2012010101	201201	국어	20120101	20120630	20120101
2012010102	201201	국어	20120101	20120630	20120101
2012010103	201201	국어	20120101	20120630	20120101
2012010104	201201	국어	20120101	20120630	20120101
2012010105	201201	국어	20120101	20120630	20120101
2012010106	201201	국어	20120101	20120630	20120101
2012010107	201201	국어	20120101	20120630	20120101
2012010108	201201	국어	20120101	20120630	20120101
2012010109	201201	국어	20120101	20120630	20120101
2012010110	201202	영어	20120101	20120710	20120101
2012010111	201202	영어	20120101	20120710	20120101
2012010112	201202	영어	20120101	20120710	20120101
2012010113	201202	영어	20120101	20120710	20120101
2012010114	201202	영어	20120101	20120710	20120101
2012010115	201202	영어	20120101	20120710	20120101
2012010116	201202	영어	20120101	20120710	20120101
2012010117	201203	수학	20120201	20120803	20120101
2012010118	201203	수학	20120201	20120803	20120101
2012010119	201203	수학	20120201	20120803	20120101
2012010120	201203	수학	20120201	20120803	20120101

위의 예제에서 과목명을 보면 과목코드에 종속은 되나 학번에는 종속되지 않는다는 것을 확인할 수 있다. 이러한 속성(과목명, 수강기간)들은 과목코드를 UID로 하는 부모 엔티티로 분리하여 새로 만들어 준다. 제 2 정규화를 진행하고 나면 중복되어 등록되어지는 데이터 항목들이 따로 분리되어 만들어지면서 중복되지 않는 데이터를 구성을 하여 데이터 관리의 효율성을 높일 수 있다.

## 정규화된 후의 데이터구조

학과등록 테이블

학번	과목코드	등록일자
2012010101	201201	20120101
2012010102	201201	20120101
2012010103	201201	20120101
2012010104	201201	20120101
2012010105	201201	20120101
2012010106	201201	20120101
2012010107	201201	20120101
2012010108	201201	20120101
2012010109	201201	20120101
2012010110	201202	20120101
2012010111	201202	20120101
2012010112	201202	20120101
2012010113	201202	20120101
2012010114	201202	20120101
2012010115	201202	20120101
2012010116	201202	20120101
2012010117	201203	20120101
2012010118	201203	20120101
2012010119	201203	20120101
2012010120	201203	20120101

과목 테이블

과목코드	과목명	수강시작일자	수강종료일자
201201	국어	20120101	20120630
201202	영어	20120101	20120710
201203	수학	20120201	20120803

### 제 2 정규형 제약조건

- 어떤 테이블이 제 1 정규형이고, 키에 속하지 않는 속성 모두가 기본 키에 완전 함수 종속이어야 한다. 즉 1차 정규화가 끝난 상태에서 기본키 열과 기본키가 아닌 열간의 종속관계가 있을 경우 이를 별도의 테이블로 분리한 것이다.

등록번호	등록일	학생번호	학생이름	과목번호	학점	성적
1	2006.2.20	1	김태희	1	2	90
1	2006.2.20	1	김태희	2	2	83
2	2006.2.21	2	한채영	1	2	95
3	2006.2.22	1	김태희	3	3	75

과목

과목번호	학점
1	2
2	2
3	3

등록-과목

등록번호	과목번호	성적
1	1	90
1	2	83
2	1	95
3	3	75

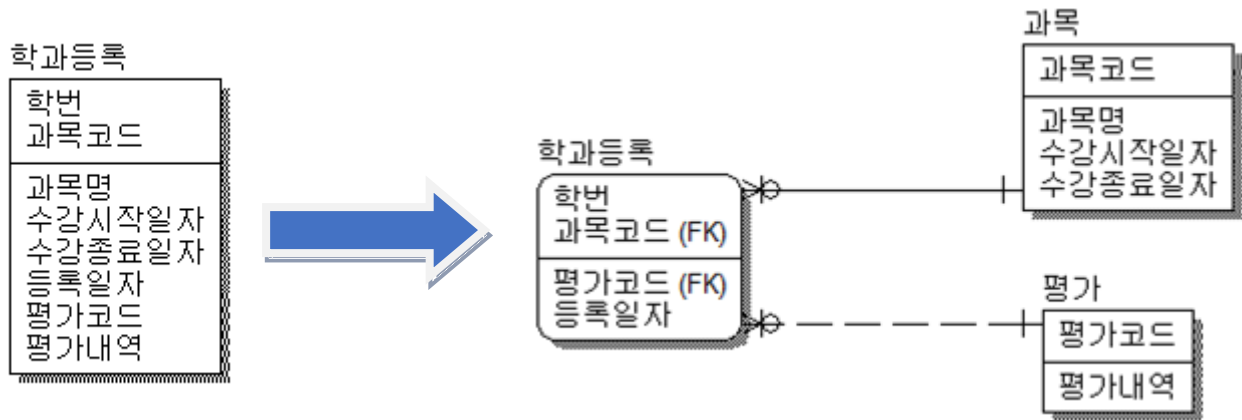
등록-학생

등록번호	등록일	학생번호	학생이름
1	2006.2.20	1	김태희
2	2006.2.21	2	한채영
3	2006.2.22	1	김태희

### 제 3 정규화

식별자 이외의 속성은 식별자가 아닌 다른 속성에 종속적이지 않아야 한다.

- UID가 아닌 모든 속성 간에는 서로 종속될 수 없다. (속성 간 종속성 배제)
- 정규화가 이루어지고 나면 부모 엔티티로 분리된다.



위의 예제에서 평가코드와 평가 내역을 보면 평가내역이 평가코드에 종속적이라는 것을 알 수 있다. 속성값(평가코드)에 종속된다는 것이다.

이런 경우에 평가코드를 UID로 하고 평가내역을 속성값으로 하는 엔티티로 분리하여 새로 만들어 준다. 3정규화를 진행하고 나면 2정규화와 유사하게 중복되어 등록되어지는 데이터 항목들이 따로 분리되어 만들어 진다.

### 정규화되기 전의 데이터구조

학과등록 테이블

학번	과목코드	등록일자	평가코드	평가내역
2012010101	201201	20120101	A	A 평가내역입니다.
2012010102	201201	20120101	A	A 평가내역입니다.
2012010103	201201	20120101	A	A 평가내역입니다.
2012010104	201201	20120101	A	A 평가내역입니다.
2012010105	201201	20120101	A	A 평가내역입니다.
2012010106	201201	20120101	A	A 평가내역입니다.
2012010107	201201	20120101	A	A 평가내역입니다.
2012010108	201201	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010109	201201	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010110	201202	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010111	201202	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010112	201202	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010113	201202	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010114	201202	20120101	B	B 평가내역을 기술하는부분 입니다.
2012010115	201202	20120101	C	C 평가내역을 기술하는부분 입니다.
2012010116	201202	20120101	C	C 평가내역을 기술하는부분 입니다.
2012010117	201203	20120101	C	C 평가내역을 기술하는부분 입니다.
2012010118	201203	20120101	C	C 평가내역을 기술하는부분 입니다.
2012010119	201203	20120101	C	C 평가내역을 기술하는부분 입니다.
2012010120	201203	20120101	C	C 평가내역을 기술하는부분 입니다.

## 정규화된 후의 데이터구조

학과등록 테이블

학번	과목코드	등록일자	평가코드
2012010101	201201	20120101	A
2012010102	201201	20120101	A
2012010103	201201	20120101	A
2012010104	201201	20120101	A
2012010105	201201	20120101	A
2012010106	201201	20120101	A
2012010107	201201	20120101	A
2012010108	201201	20120101	B
2012010109	201201	20120101	B
2012010110	201202	20120101	B
2012010111	201202	20120101	B
2012010112	201202	20120101	B
2012010113	201202	20120101	B
2012010114	201202	20120101	B
2012010115	201202	20120101	C
2012010116	201202	20120101	C
2012010117	201203	20120101	C
2012010118	201203	20120101	C
2012010119	201203	20120101	C
2012010120	201203	20120101	C

평가 테이블

평가코드	평가내역
A	A 평가내역입니다.
B	B 평가내역을 기술하는부분 입니다.
C	C 평가내역을 기술하는부분 입니다.

### 제 3 정규형 제약조건

과목

과목번호	학점
1	2
2	2
3	3

등록-과목

등록번호	과목번호	성적
1	1	90
1	2	83
2	1	95
3	3	75

과목

과목번호	학점
1	2
2	2
3	3

등록-과목

등록번호	과목번호	성적
1	1	90
1	2	83
2	1	95
3	3	75

등록-학생

등록번호	등록일	학생번호	학생이름
1	2006.2.20	1	김태희
2	2006.2.21	2	한채영
3	2006.2.22	1	김태희

등록-학생

등록번호	등록일	학생번호
1	2006.2.20	1
2	2006.2.21	2
3	2006.2.22	1

학생

학생번호	학생이름
1	김태희
2	한채영

주식별자 정의는 전체 데이터 모델의 복잡성을 결정하는 중요한 요소이다.

- 해당 업무에서 자주 이용되는 속성을 주 식별자로 지정한다.
- 속성값의 길이가 가변적인 속성은 주식별자로서 적당하지 않다. 부서이름보다는 부서코드를 주식별자로 지정하라.
- 속성값이 자주 변하는 속성은 주식별자로서 적당하지 않다.
- 주식별자를 선정하기 위한 속성(컬럼)의 수를 적게 한다.
- 주식별자에는 NULL 데이터가 들어와서는 안된다.