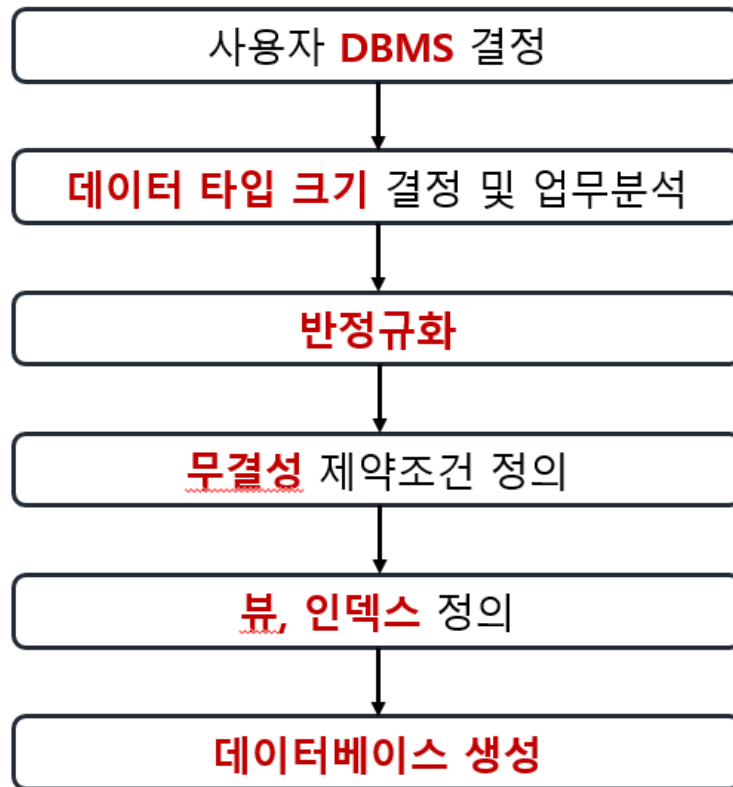


@1. 물리모델

1-1. 물리 모델이란



- 실제로 데이터베이스에 이식할 수 있도록 모델의 구조보다는 인덱스, 뷰, 테이블 타입과 사용하는 DBMS와 같은 물리적 요소에 집중하여 데이터베이스를 설계하는 단계이다.
- 성능을 고려해 비정규화(Denormalization)를 진행하는 단계이다.

논리적 DB 설계 (데이터 모델링)	물리적 DB 설계
DBMS의 종류나 제품에 상관없이 진행 (ERD는 어떤 데이터베이스를 사용해도 적용 가능)	특정 DBMS를 전제로 진행 (적용 DBMS의 특성을 고려)

1-2. 물리 모델의 목적

- 성능을 최적화 하기 위해 성능을 고려하여 엔티티를 합치거나 모델 구조를 약간 변경(비정규화)할 수 있다.

1-3. 물리 모델의 주의사항

- 모델의 구조는 많아도 10% 이내의 변화 정도만 이루어 져야 한다.

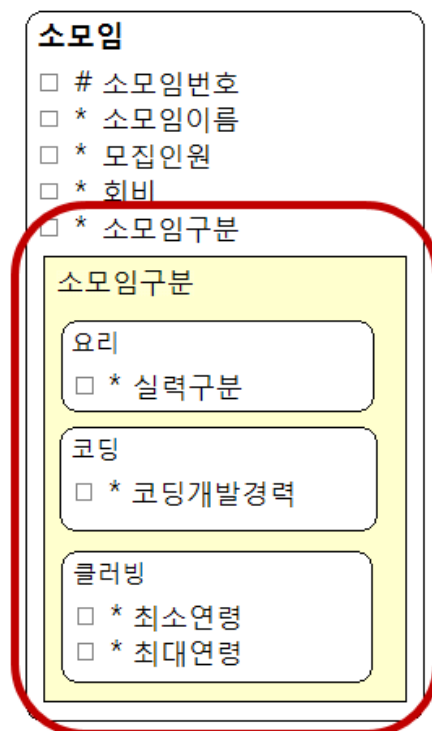
- 화면이나 SQL이 작성되면서 필요한 인덱스와 뷰를 도출한다.
- 비정규화는 특정 성능 문제를 해결하기 위한 목적이 아니라면 고려하지 않는다.
- 슈퍼타입과 서브관계의 물리적 변환

2. 슈퍼타입과 서브타입

2-1. 슈퍼타입과 서브타입이란

요리소모임 <ul style="list-style-type: none"> <input type="checkbox"/> # 요리소모임번호 <input type="checkbox"/> * 요리소모임이름 <input type="checkbox"/> * 모집인원 <input type="checkbox"/> * 회비 <input type="checkbox"/> * 실력구분 	코딩소모임 <ul style="list-style-type: none"> <input type="checkbox"/> # 코딩소모임번호 <input type="checkbox"/> * 코딩소모임이름 <input type="checkbox"/> * 모집인원 <input type="checkbox"/> * 회비 <input type="checkbox"/> * 코딩개발최소경력 	클러빙소모임 <ul style="list-style-type: none"> <input type="checkbox"/> # 클러빙소모임번호 <input type="checkbox"/> * 클러빙소모임이름 <input type="checkbox"/> * 모집인원 <input type="checkbox"/> * 회비 <input type="checkbox"/> * 최소연령 <input type="checkbox"/> * 최대연령
--	--	--

- 위의 세가지 소모임 관련 엔터티는 유사한 속성들을 지니고 있다. 공통적으로 지닌 유사한 속성들을 기준으로 하나의 엔터티로 일반화를 진행할 수 있다.



- 노란색 부분을 제외한 속성들은 공통적인 소모임 엔터티들의 공통적인 속성들을 의미하며 슈퍼타입으로 볼 수 있으며 소모임 구분이라는 속성을 통해 서브타입에 해당하는 속성들이 존재하게 된다.

- 서브타입에 해당하는 속성들은 소모임 구분에 따라 해당되는 속성이 있고 해당되지 않는 속성이 있으며 예시의 경우 각 서브타입 속성은 소모임 구분에 따라 배타적인 관계로 관리된다.
- 소모임에서 소모임 구분에 따라 슈퍼타입과 서브타입으로 속성을 구분하여 통합 엔터티로 표시한 것이지만 이전에 세개의 엔터티로 표기한 것처럼 서브타입의 갯수만큼 엔터티가 따로 존재했던 것으로 생각할 수 있다.

소모임

☐ # 소모임번호
 ☐ * 소모임이름
 ☐ * 모집인원
 ☐ * 회비
 ☐ * 소모임구분

소모임구분

요리
☐ * 실력구분

실력구분

초급

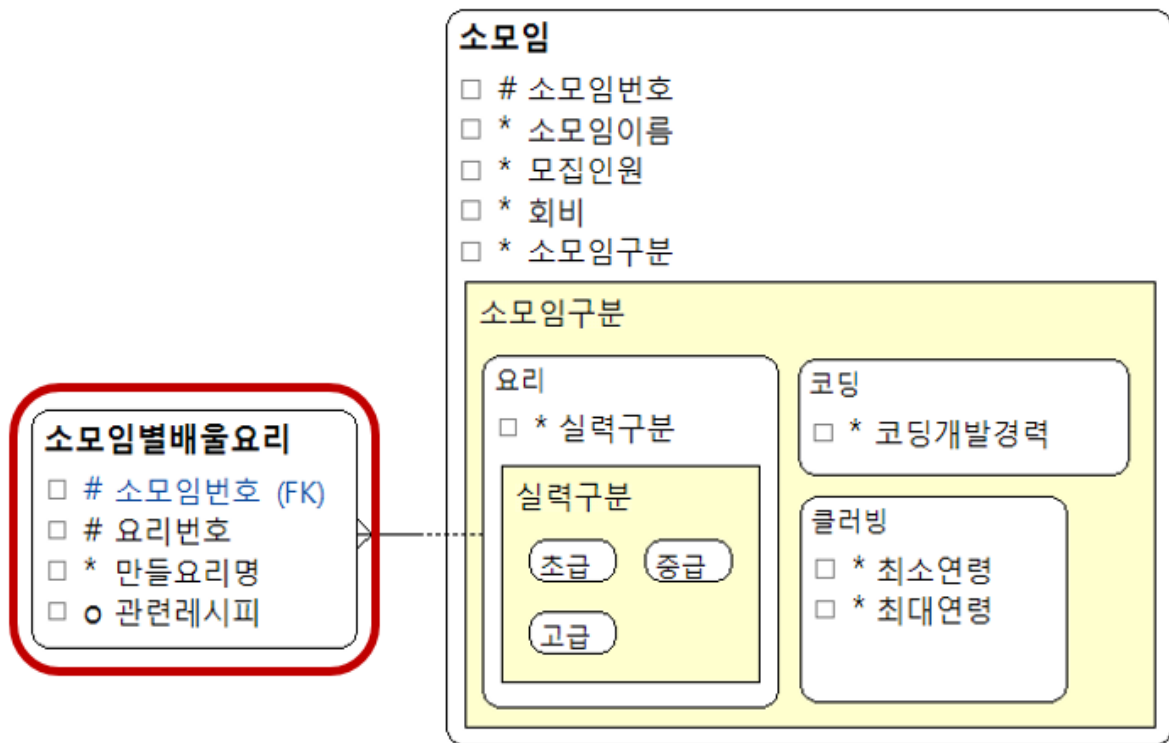
중급

고급

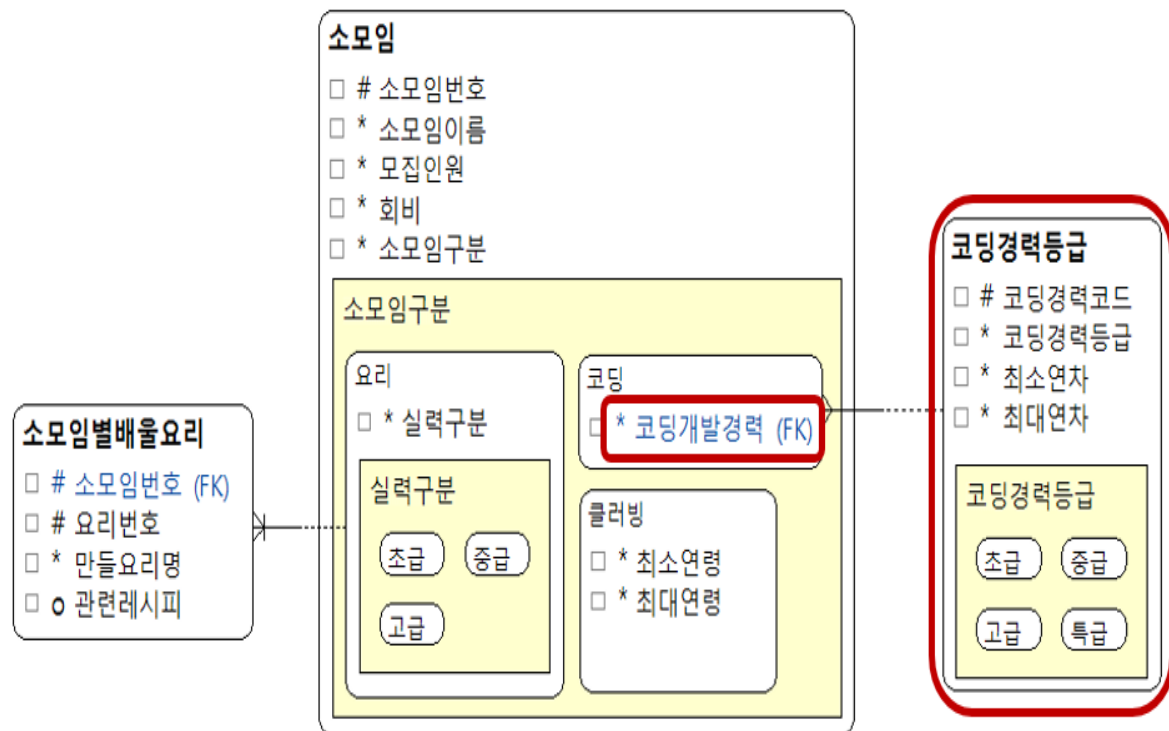
코딩
☐ * 코딩개발경력

클럽
☐ * 최소연령
☐ * 최대연령

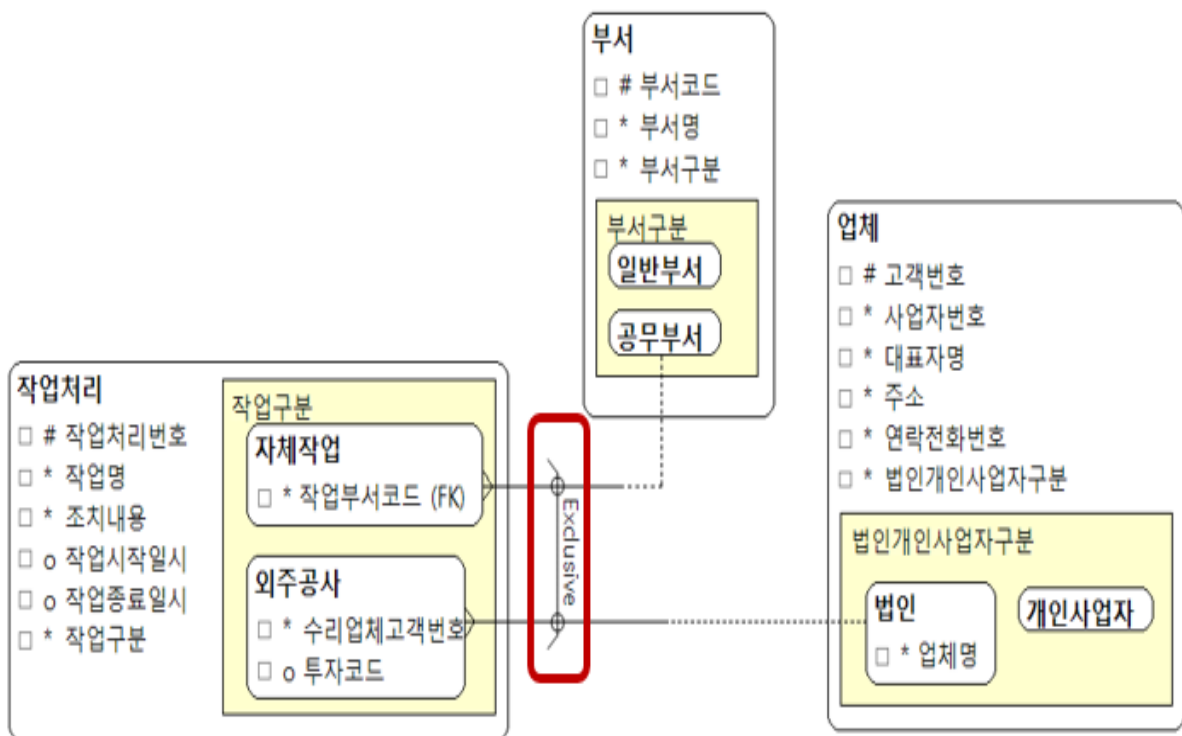
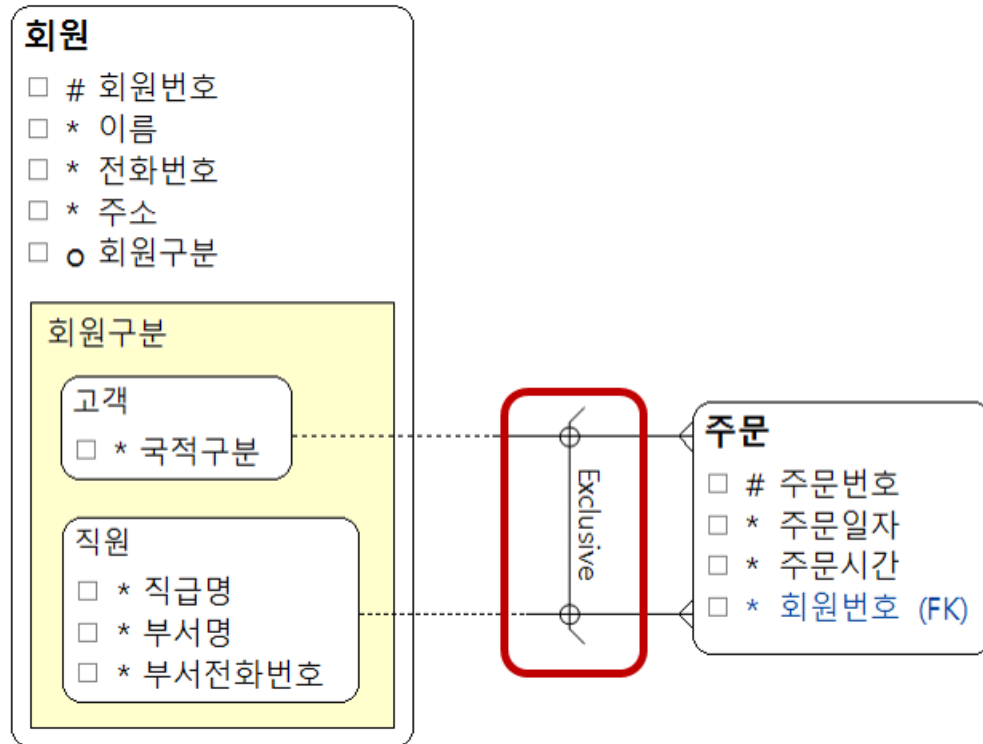
- 서브타입중에 요리에 해당하는 소모임일 실력 구분 속성에 해당하는 속성값이 셋중의 하나인 배타적관계 임을 나타낼 수 있다. 이는 실력 구분별 요리 소모임 엔터티가 초급, 중급, 고급일 때마다 각각 엔터티로 구분되어 존재할 수 있음을 의미한다.



- 요리 소모임의 경우 소모임별 배울 요리 엔터티와 관계를 가지게 되며 소모임별 배울 요리는 소모임의 서브타입 엔터티에 해당한다.



- 코딩 소모임의 경우도 코딩경력등급 엔터티와 관계를 맺어줄 수 있으며 소모임의 코딩개발경력 속성은 코딩경력등급 엔터티의 주식별자의 외래키가 되게 속성을 정의할 수도 있다.



- 서브타입은 서로 배타적(둘 중 하나만 해당)이고 서브타입 속성 값에 따라 서브타입엔티티들도 배타적으로 연결될 때 위와같이 배타적 관계 표시를 할 수 있다.

2-2 서브타입의 물리모델 변환

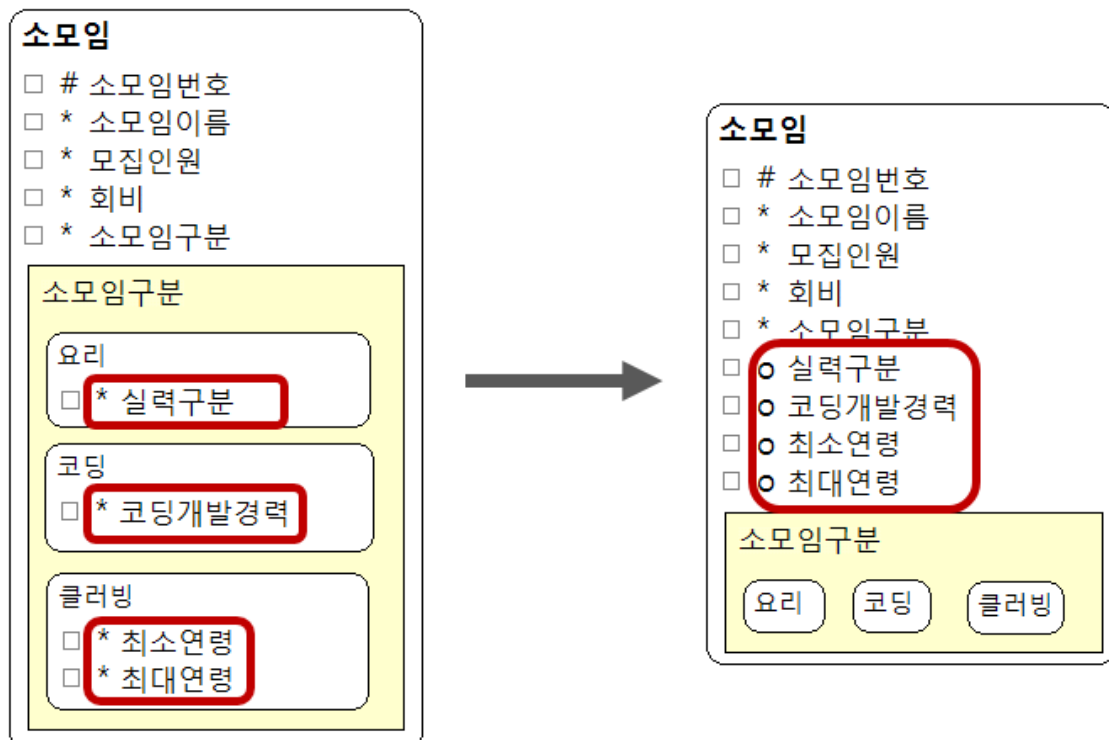
2-2-1. 통합 엔터티로 변환

장점	단점
서브타입 구분 없는 데이터 접근이 간편하다 (조회 시 조인이 필요없다.) (Entity Integrity)	테이블의 컬럼 수가 증가된다.
View를 활용해 각 서브타입 조회 및 수정이 가능하다.	서브타입 별로 처리 시 서브타입의 구분이 필요해 지는 경우가 많다.
	특정한 서브타입을 NOT NULL로 제한할 수 없다.

2-2-2. 각 서브타입마다 하나의 엔터티로 변환

장점	단점
서브타입 별로 처리 시 서브타입의 유형 구분이 불필요하다. (Entity Integrity)	전체적인 데이터를 처리하는 경우 UNION이 발생한다.
단위 테이블의 크기가 감소한다.	여러 테이블을 합친 View는 조회만 가능하며 인스턴스를 개별적으로 구분하기 위한 UID 유지 관리가 어렵다.
불필요한 컬럼이 줄어든다.	복잡한 SQL 처리 시 통합이 어렵다.

2-2-3. 서브타입 속성의 NULLABLE



- 물리모델로 변환할 때는 배타적 관계에 해당하는 서브타입의 속성들은 NULL값이 들어올 수 있음을 고려해야 한다.

3. 성능 개선

3-1. 뷰와 인덱스 설계

3-1-1. 뷰란

- 하나의 테이블, 혹은 여러 테이블에 대하여 특정 사용자나 조직의 관점에서 데이터를 바라볼 수 있도록 해주는 수단으로서 가상 테이블이라고도 부른다.
- 복잡한 쿼리가 요구 되는 것들을
- 접근 권한 제어를 위해 생성할 수 있다.

3-1-2. 인덱스란

- 인덱스는 검색의 기준이 되는 컬럼만을 뽑아 정렬한 상태를 유지하고 있으며 인덱스의 각 튜플은 원래 데이터가 저장되어 있는 테이블에 대응하는 튜플의 주소 값을 가지고 있다.
- SQL의 WHERE절에서 비교 대상이 되는 컬럼 또는 JOIN에 사용되는 컬럼이어야 한다.
- 튜플의 수가 적으면 인덱스를 지정하여도 별 효과가 없다.
- 인덱스로 지정한 컬럼에 의해 검색했을 때 검색 결과가 전체 튜플의 10~15% 미만일 때 인덱스의 효과가 있다.
- 기본키로 지정한 컬럼에 대해서는 자동적으로 인덱스를 만들어 준다.

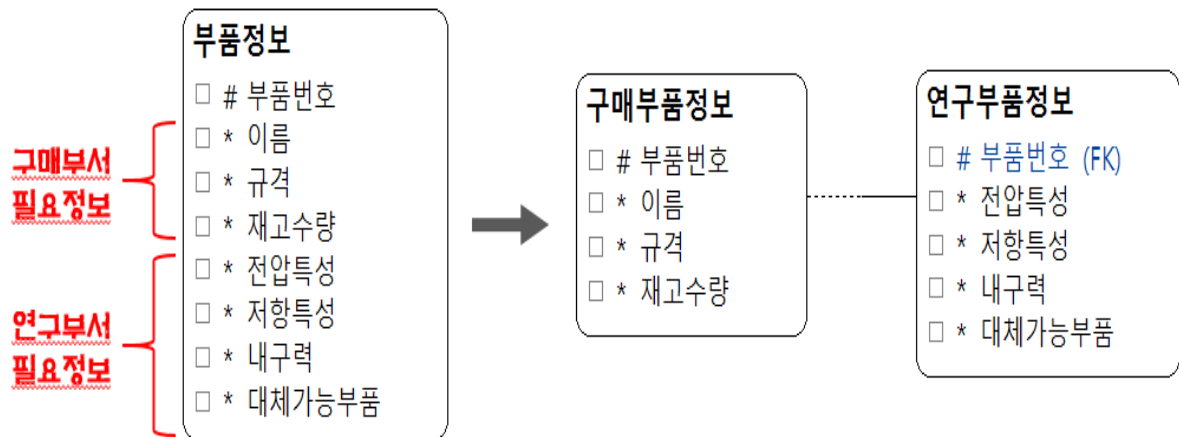
3-2. 비정규화(Denormalization)

3-2-1. 비정규화란

- 정규화 작업이 완료된 후 데이터 물리 모델링 과정 중 시스템의 성능 향상, 개발 과정의 편의성, 운영의 단순화를 추구
- 중복은 감수하고 데이터베이스의 성능을 향상시키는 것. (특히 검색 속도)
- 정규화를 통한 데이터 무결성 유지도 중요하지만, 다수 사용자가 동시 이용하는 환경에서 일정 성능을 유지하는 것도 매우 중요.

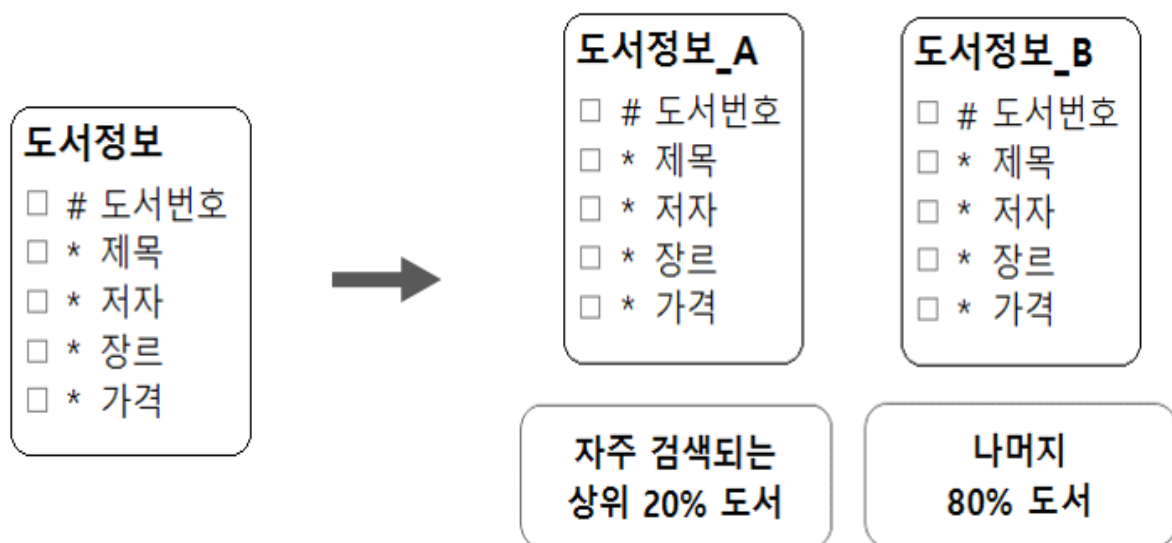
3-2-2. 비정규화의 종류

3-2-2-1. 수직 분할



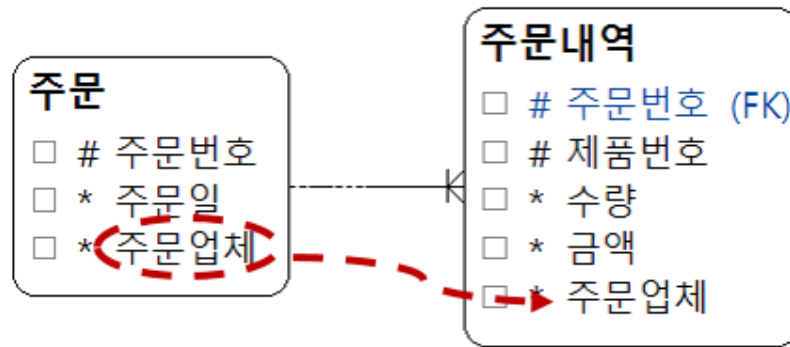
- 엔터티의 튜플 수 및 속성의 수가 매우 많고, 엔터티의 속성들이 그룹화되어 각 그룹이 특정 부서 혹은 응용 프로그램에 의해서만 사용될 때
- 엔터티의 데이터량이 많을수록 검색속도는 느려지므로, 엔터티를 분할하여 데이터량을 줄임으로써 성능 향상을 도모할 수 있다.
- 수직 분할은 반복되지는 않지만 속성들이 그룹화 되어 각 그룹이 특정 부서 혹은 응용프로그램에 의해서만 사용될 때에 사용할 수 있으며 수직 분할하면 본래 하나의 엔터티였기 때문에 카디널리티는 1:1이 된다.

3-2-2-2. 수평 분할



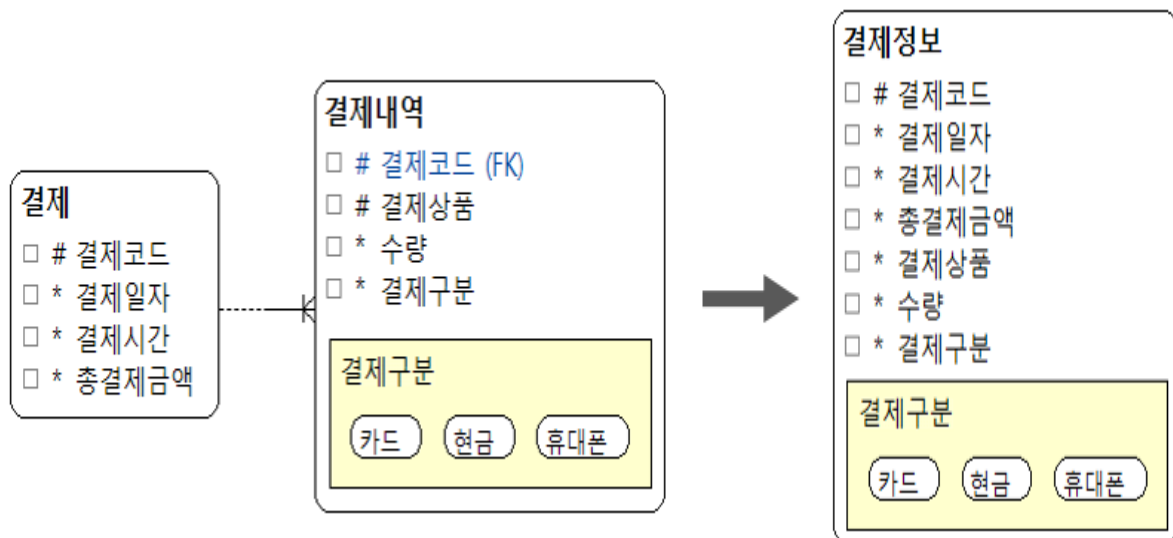
- 튜플의 검색빈도가 다르다는 점을 이용하여 엔터티를 분할하면 엔터티의 데이터 크기가 감소하여 성능 향상의 효과를 볼 수 있다.
- 수평 분할 결과로 나온 엔터티 중 도서정보_A는 자주 검색되는 상위 20%의 도서이고 도서정보_B는 나머지 80%의 도서이다. 즉, 20%의 책이 전체 검색 빈도의 80%를 차지한다고 생각할 수 있으며 수평분할 된 엔터티는 속성은 동일하지만 서로 아무런 관계도 성립하지 않는다.

3-2-2-3. 속성 중복



- 주문 내역에 주문업체 속성이 중복되게 추가한 것으로 조인을 하지 않고 주문내역으로부터 주문업체 정보를 얻을 수 있다.
- 과도하게 중복된 속성들을 사용하게 되면 데이터의 무결성(데이터의 불일치)이 깨질 수 있다.
- 입력, 수정, 삭제에 있어 응답시간이 늦어질 수 있다.

3-2-2-4. 엔터티 통합



- 항상 혹은 대부분 조인에 의한 검색을 하고, 검색이 빈번히 이루어지는 두 개 이상의 엔터티를 대상으로 한다.
- 조인 연산에 걸리는 시간을 단축시켜 준다.
- 중복이 존재할 수 있어 이상(Anomaly) 현상이 발생할 수 있다는 것을 고려해야 한다.

4. 테이블 타입

4-1. 문자 데이터 타입

데이터타입	설명
-------	----

CHAR(n)	고정길이 문자 / 최대 2000byte / 디폴트 값은 1byte
VARCHAR2(n)	가변길이 문자 / 최대 4000byte / 디폴트 값은 1byte
NCHAR(n)	고정길이 유니코드 문자(다국어 입력가능) / 최대 2000byte / 디폴트 값은 1byte
NVARCHAR(n)	가변길이 유니코드 문자(다국어 입력가능) / 최대 2000byte / 디폴트 값은 1byte
LONG	최대 2GB 크기의 가변길이 문자형
CLOB	대용량 텍스트 데이터 타입(최대 4Gbyte)
NCLOB	대용량 텍스트 유니코드 데이터 타입(최대 4Gbyte)

4-2. 숫자형 데이터 타입

데이터타입	설명
NUMBER(P,S)	가변숫자 / P(1 ~ 38, 디폴트 값: 38) / S(-84 ~ 127, 디폴트 값: 0) / 최대 22byte
FLOAT(P)	NUMBER의 하위타입 / P(1 ~ 128, 디폴트 값: 128) / 이진수 기준 / 최대 22byte
BINARY_FLOAT	32비트 부동소수점 수 / 최대 4byte
BINARY_DOUBLE	64비트 부동소수점 수 / 최대 8byte

4-3. 날짜 데이터 타입

데이터타입	설명
DATE	BC 4712년 1월 1일부터 9999년 12월 31일, 연, 월, 일, 시, 분, 초까지 입력 가능
TIMESTAMP	연도, 월, 일, 시, 분, 초, 밀리초까지 입력 가능

4-4. LOB 데이터 타입

데이터타입	설명
CLOB	문자형 대용량 객체 고정길이와 가변길이 문자집합 지원
NCLOB	유니코드를 지원하는 문자형 대용량 객체
BLOB	이진형 대용량 객체
BFILE	대용량 이진 파일에 대한 위치, 이름 저장