

## 2. 정규화

### 정규화의 정의

- 관계형 데이터베이스를 설계 시 중복과 갱신 이상을 최소화하기 위해서 데이터를 구조화 하는 작업!
- 정규화를 해야하는 이유?

이름	회원번호	스터디명	스터디번호	스터디 생성일	...
신짱구	1	리팩토링	1	10월9일	...
신짱구	1	짱구의 일기	2	1월16일	...
신짱구	1	디우 괴롭히기	3	11월15일	...
김디우	2	리팩토링	1	10월9일	...
김디우	2	짱구의 일기	2	1월16일	...

스터디 관리 서비스를 만든다하자. 위와 같이 테이블을 설계한다면 어떤 문제가 발생할까?

- **문제점** : 한 릴레이션에 여러 엔티티의 애트리뷰트들이 존재하면서 정보가 중복 저장된다.
  - 저장 공간 낭비
  - 갱신 이상 발생
    - **삽입 이상** : 불필요한 정보를 함께 저장하지 않고는 어떤 정보를 저장하는 것이 불가능하다.
      - ex) 예시에서 회원만 추가 하는 것이 불가능
    - **삭제 이상** : 어떤 정보를 삭제하려면 유용한 정보까지 함께 삭제 되는 문제
      - ex) 신짱구를 삭제하면 디우 괴롭히기 스터디도 함께 소멸된다.
    - **수정 이상** : 어떤 정보를 수정할 때, 같은 정보지만 다른 데이터는 수정되지 않아 데이터의 불일치가 발생

- ex) 첫 번째 신짱구 데이터의 이름을 흰둥이로 바꿔도 나머지 신짱구 데이터는 여전히 신짱구로 남아있음

이러한 문제점들을 해결하기 위해 정규화를 진행한다!

## 정규화의 과정

### 제 1 정규화

- 릴레이션의 모든 애트리뷰트가 원자값만을 갖는다.
- 즉 반복그룹이 나타나면 안된다.

상품코드	상품명	주문 수량
a1a1a1, b2b2b2, c3c3c3	액션가면, 초코비, 흰둥이 사료	만개, 천개, 백개
d4d4d4, f5f5f5	밥그릇, 뼈다귀	오만개, 오천개
		...

상품코드	상품명	주문 수량
a1a1a1	액션가면	만개
b2b2b2	초코비	천개
c3c3c3	흰둥이 사료	백개
d4d4d4	밥그릇	오만개
f5f5f5	뼈다귀	오천개

→ 하나의 애트리뷰트에 대해 중복되는 값을 튜플로 분리

### 제 2 정규화

이름	회원번호	스터디명	스터디번호	스터디 생성일	직책
----	------	------	-------	---------	----

- 완전 함수적 종속성
  - 예시에서 직책은 회원번호, 스터디번호에 종속 → 두 기본키에 대해 완전히 종속적이고 이를 완전 함수적 종속성이라 한다.
- 부분 함수적 종속성

- 예시에서 이름은 회원번호에만 종속적이며, 스터디 명, 스터디 생성일은 스터디 번호에만 종속적이다. 이를 부분 함수적 종속성이라 한다.
- 제1 정규형을 만족한 테이블에 대해서 **완전 함수 종속**을 만족하도록 테이블을 분리하는 것.

학번	이름	소속학과	학과장
1901	홍길동	컴퓨터과	김교수
1845	임걱정	전자과	박교수
1706	신사임당	기계과	천교수
1925	장영실	컴퓨터과	김교수

- 위 예시에서 기본키인 학번에 대해, 이름과 소속학과는 학번에 완전 종속 된다. 단, 학과장은 소속학과만으로 종속되므로 **부분적인 종속**관계가 된다. 이를 분리해주는게 제 2 정규화이다.

학번	이름	소속학과
1901	홍길동	컴퓨터과
1845	임걱정	전자과
1706	신사임당	기계과
1925	장영실	컴퓨터과

학과	학과장
컴퓨터과	김교수
전자과	박교수
기계과	천교수

## 제 3 정규화

- 이행 함수적 종속성

주문번호	주문일자	회원번호	회원이름
------	------	------	------

- 릴레이션에서 X, Y, Z라는 3개의 속성이 있을 때  $X \rightarrow Y$ ,  $Y \rightarrow Z$  이란 종속 관계가 있을 경우  $X \rightarrow Z$ 가 성립될 때를 말한다. 즉, X를 알면 Y를 알고 그를 통해 Z를 알 수

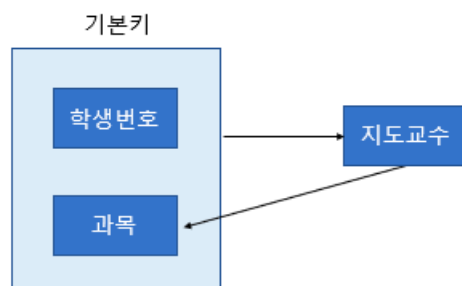
있는 경우

- 예시에서는 주문번호를 알면 회원번호를 알 수 있고, 이를 통해 회원 이름을 알 수 있다.
- 제2 정규화를 진행한 테이블에 대해 **이행적 종속**을 없애도록 테이블을 분해하는 것이다.
  - 위 예시에서는 주문번호 주문일자 회원번호 / 회원번호 회원이름 으로 분리

## 보이스-코드 정규화(BCNF)

- 제 3 정규형을 만족하며, 어떤 애트리뷰트 값을 결정하는 애트리뷰트(**결정자**)가 모두 후보키 집합에 속해야한다.

학생번호	과목	지도교수
101	자바	김자바
101	C++	박플플
102	자바	오자바
103	C#	조씨샵
104	자바	김자바



위의 테이블은 (학생번호, 과목)이 기본키로 지도교수를 알 수 있다. 그러나 지도교수를 알면 과목을 알 수 있으므로, 지도교수 → 과목 이 종속적이다.

즉, 후보키 집합에 속하지 않은 결정자가 존재하므로 BCNF를 만족하지 않는다.

BCNF를 만족하기 위해서는 다음과 같이 분리하면 된다.

학생번호	지도교수
101	김자바
101	박플플
102	오자바
103	조씨샵
104	김자바

지도교수	과목
김자바	자바
박플플	C++
오자바	자바
조씨샵	C#



## 제 4 정규화

- BCNF를 만족하며 **다치종속(다중값 종속)**을 없애야한다.
- 다치종속
  - 한 릴레이션에서 (A) B,C 일 때 B와 C가 무관한 경우

이름	과목	동아리
홍길동	데이터베이스	농구
홍길동	보안	축구
홍길동	데이터베이스	야구

위와 같이 관계가 있을 경우, 과목을 추가할 경우 동아리도 함께 추가해줘야 한다.

이처럼 B,C가 서로 상관없지만 같이 묶여있는 경우 다치 종속이라 한다.

- 위 예에서 이름, 과목 / 이름, 동아리 와 같이 릴레이션을 분리하여 다치종속을 제거해줘야 한다.

## 제 5 정규화

- 제 4정규화를 만족하며 조인 종속성을 없애야한다. 즉 더 이상 비손실 분해를 할 수 없어야 한다.
- 조인종속
  - 하나의 릴레이션을 여러개의 릴레이션으로 분해하였다가, 다시 조인했을 때 데이터 손실이 없고 필요없는 데이터가 생기는 것을 말한다. 조인 종속성은 다치 종속의 개념을 더 일반화한 것이다.
  - 원본

개발자	자격증	언어
홍길동	정보처리기사	C
홍길동	빅데이터 분석기사	C++
장길산	정보보안기사	JAVA

개발자	자격증
홍길동	정보처리기사
홍길동	빅데이터 분석기사
장길산	정보보안기사

자격증	언어
정보처리기사	C
빅데이터 분석기사	C++
정보보안기사	JAVA

개발자	언어
홍길동	C
홍길동	C++
장길산	JAVA

위처럼 3개의 테이블로 분리하여 조인을 했을 때 원본 테이블이 나올 수 있어야 한다.