

### 3. 기본 정규형과 정규화 과정

#### 정규형(Normal Form)

릴레이션(=변수들의 관계)이 정규화된 정도

#### 분류

- 크게 6가지로 분류
- 기본정규형** : 제 1~3정규형, 보이스/코드 정규형
- 고급정규형** : 제 4~5 정규형

#### 특성

- 정규형 차수가 **높아질수록**
  - 요구되는 제약조건 많아지고 엄격
  - 릴레이션 분해 많아짐 -> 데이터 중복이 줄어들
  - 이상 현상**이 발생하지 않는 바람직한 릴레이션이 됨

#### 정규화

함수 종속성을 이용해 릴레이션을 연관선이 있는 속성들로만 구성되도록 **분해**하는 것

- 목적
  - 이상 현상이 발생하지 않는 바람직한 릴레이션으로 만들어가는 과정
  - 불필요한 데이터를 제거
- 특성
  - 무손실 분해 : 정보손실 발생하면 안됨
  - 릴레이션은 **의미적으로 동등한** 릴레이션들로 분해되어야함
  - 분해된 릴레이션을 **자연조인**하면 **분해 전 릴레이션으로 복원가능**해야함

#### 1~3 정규형

##### 제 1 정규형(1NF)

- 릴레이션에 속한 모든 속성의 도메인이 **원자값(not tuple, list)**으로만 구성
- 제 1정규형을 만족해야 관계 **데이터베이스의 릴레이션이 될 자격**이 있음

<u>고객아이디</u>	<u>이벤트번호</u>	당첨여부	등급	할인율
apple	E001, E005, E010	Y, N, Y	gold	10%
banana	E002, E005	N, Y	vip	20%
carrot	E003, E007	Y, Y	gold	10%
orange	E004	N	silver	5%

\*(밀줄친 값들(고객아이디, 이벤트번호)은 기본키)

이벤트번호, 당첨여부가 다중값 가지고있다. 제 1정규형 만족X

-> 아래와 같이 변환해야함

고객아이디	이벤트번호	당첨여부	등급	할인율
apple	E001	Y	gold	10%
apple	E005	N	gold	10%
apple	E010	Y	gold	10%
banana	E002	N	vip	20%
banana	E005	Y	vip	20%
carrot	E003	Y	gold	10%
carrot	E007	Y	gold	10%
orange	E004	N	silver	5%

1정규형은 만족하지만 여전히 이상(삽입, 갱신, 삭제)현상 발생

- 삽입이상 : id가 melon인데 이벤트번호가 없으면 이벤트번호가 NULL이라 삽입불가(이벤트번호는 기본키이기 때문에)
- 갱신이상: apple등급을 gold에서 vip로바꾼다고 했을 때, 3개 중 2개만 바뀌버리면 문제 발생 (apple의 등급은 gold? or vip?)
- 삭제이상 : orange 지우면 silver 등급의 할인율 정보도 삭제됨(위 table에서 silver 등급은 orange 에 밖에 없기때문에) - 정보손실

## 제2 정규형(2NF)

- 릴레이션이 제1정규형에속하고, 기본키가 아닌 모든 속성이 기본키에 **완전 함수종속**되면 제2정규형 만족

### 참고

**완전 함수종속** : 종속자가 기본키에만 종속되는것. 기본키가 여러개라면 종속자는 여러개의 기본키 특성에 모두 종속이 되어야함

위에서 **당첨여부** 는 **이벤트번호** 에만 종속되고, **등급** 은 **고객아이디** 에만 종속되므로 완전함수종속 관계가 아니다.

고객ID	상품코드	주문상품	수량	가격
AAAA01	T001	티셔츠	2	12000
AAAA01	B110	청바지	1	11000
AAAA02	B110	청바지	2	22000
AAAA03	T091	와이셔츠	1	15000
AAAA03	O100	원피스	1	19000

이 경우 **수량** 은 **상품코드** 와 **고객ID** 두 기본키에 의해 결정되므로 완전함수종속관계임

다시 돌아가서 어떻게 제 2정규형을 만족시킬것인가? -> 고객 릴레이션과 이벤트 참여 릴레이션으로 분리

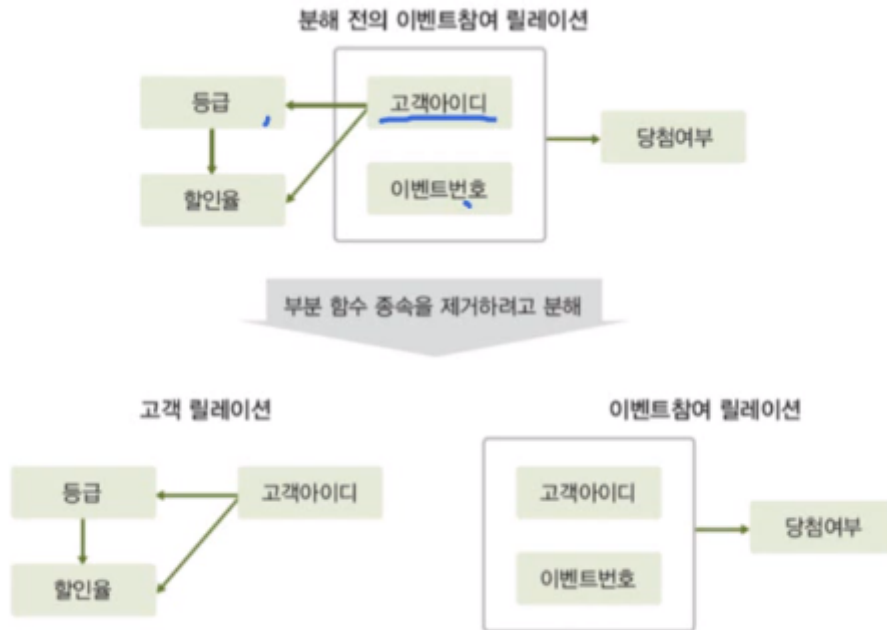


그림 9-25 고객 릴레이션과 이벤트참여 릴레이션의 함수 종속 다이어그램

But 여전히 이상현상 발생

고객아이디	등급	할인율
apple	gold	15%
banana	vip	20%
carrot	gold	10%
orange	silver	5%
NULL	bronze	1%

← 데이터 불일치로 인한 갱신 이상

← 데이터 손실로 인한 삭제 이상

← 삽입 불가로 인한 삽입 이상

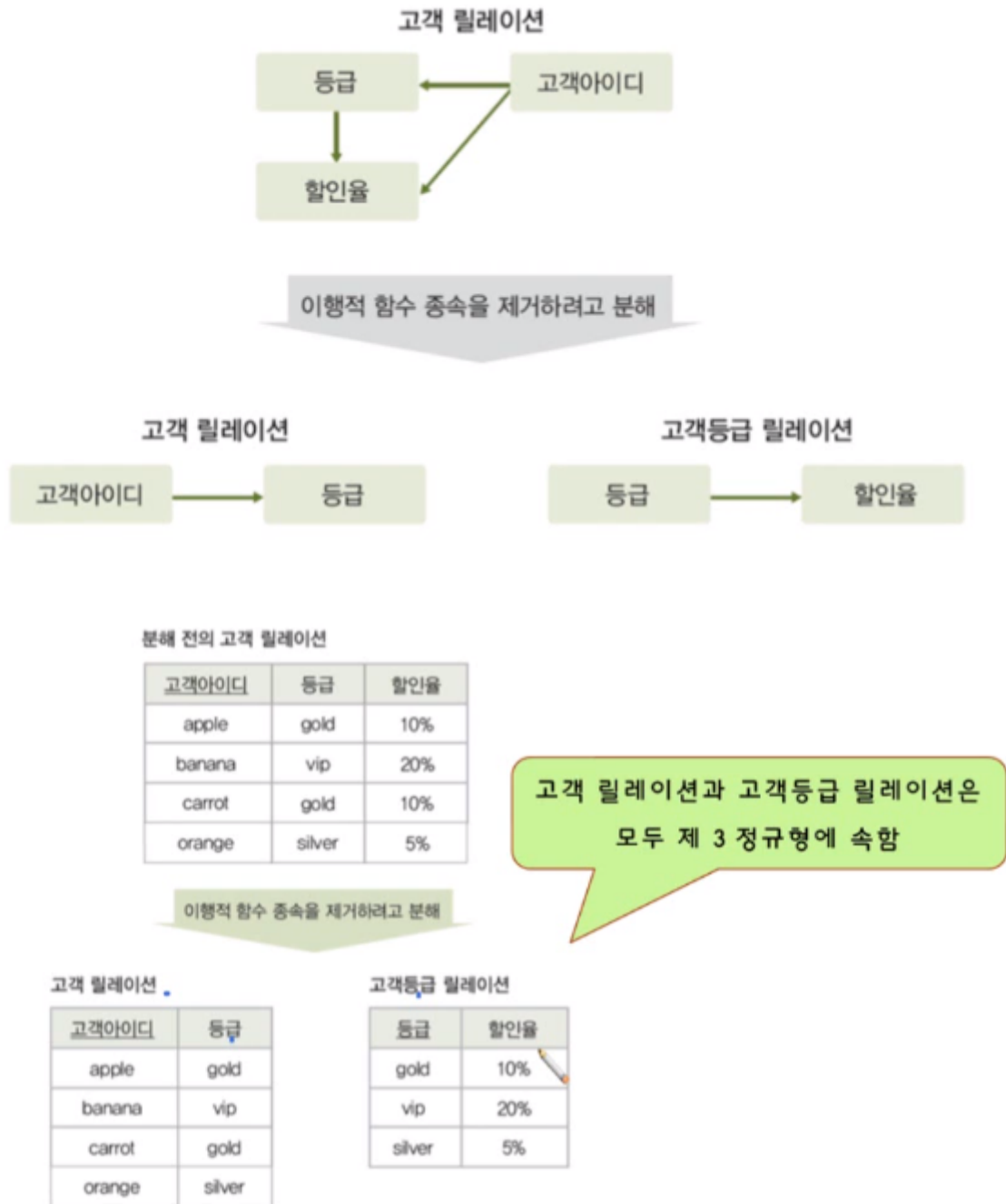
-> 제 2 정규형을 만족하더라도 이상현상이 해결되는것은 아니다.

Why? : **이행적 함수 종속**이 존재하기 때문(고객아이디가 등급결정, 등급이 할인율 결정하므로 고객아이디가 할인율을 결정할 수 있다)

따라서 이행적 함수 종속이 제거되도록 **고객** 릴레이션 분해

### 제 3 정규형(3NF)

- 릴레이션이 제 2정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 **이행적 함수 종속되지 않음**



## BCNF(Boyce Codd Normal Form) 정규형

- 3차정규형을 만족하면서 모든 결정자가 후보키 집합에 속한 정규형

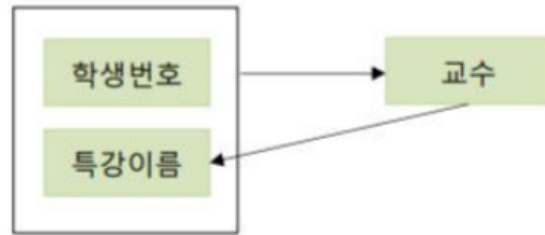
### 참고

**후보키** : 릴레이션을 구성하는 속성들 중에서 튜플을 **유일하게 식별**하기 위해 사용하는 속성들의 부분 집합, 즉 **기본키로 사용할 수 있는 속성들**을 말한다. 기본키이면 후보키이다

ex) 마을(마을 ID, 도시명, 도명)이라는 관계의 경우 마을 ID와 {도시명, 도명}이 후보 키이다. 또한 도시명에는 중복이 있을 수 있기 때문에 도시명은 단독으로는 후보 키가 되지 않는다.

### 특강수강

학생번호	특강이름	교수
501	소셜네트워크	김교수
401	소셜네트워크	김교수
402	인간과 동물	송교수
502	창업전략	박교수
501	창업전략	홍교수



\*위 관계에서 교수는 1개의 수업만 담당한다고 가정

위에서 **학생번호**, **특강이름** 두개의 기본키를 사용해야 **교수**를 종속시킬 수 있다. 또한 **교수**도 **특강이름**을 결정하기때문에 **결정자**가 된다.

**학생번호**, **특강이름**은 기본키이기때문에 후보키이지만 **교수**는 후보키가 아니므로 위 관계는 BCNF 정규형을 만족하지 못한다.

제 3정규형은 이상현상이 발생가능하며 더 엄격한 BCNF정규화를 통해 이를 해결할 수 있다.