

2025년 상반기 K-디지털 트레이닝

정규화

[KB] IT's Your Life

✓ 이상(anomaly) 현상

- 불필요한 데이터 중복으로 인해 릴레이션에 대한 데이터 삽입·수정·삭제 연산을 수행할 때 발생할 수 있는 부작용

✓ 정규화

- 이상 현상을 제거하면서 데이터베이스를 올바르게 설계해 나가는 과정

✓ 이상 현상의 종류

삽입 이상

새 데이터를 삽입하기 위해 불필요한 데이터도 함께 삽입해야 하는 문제

갱신 이상

중복 튜플 중 일부만 변경하여 데이터가 불일치하게 되는 모순의 문제

삭제 이상

튜플을 삭제하면 꼭 필요한 데이터까지 함께 삭제되는 데이터 손실의 문제

✓ 이상 현상을 설명하기 위한 릴레이션의 예

○ 이벤트 참여 릴레이션

<u>고객아이디</u>	<u>이벤트번호</u>	당첨여부	고객이름	등급
apple	E001	Y	정소화	gold
apple	E005	N	정소화	gold
apple	E010	Y	정소화	gold
banana	E002	N	김선우	vip
banana	E005	Y	김선우	vip
carrot	E003	Y	고명석	gold
carrot	E007	Y	고명석	gold
orange	E004	N	김용욱	silver

✓ 삽입 이상(insertion anomaly)

○ 릴레이션에 새 데이터를 삽입하려면 불필요한 데이터도 함께 삽입해야 하는 문제

- 아직 이벤트에 참여하지 않았고 아이디가 “melon”, 이름이 “성원용”, 등급이 “gold”인 신규 고객의 데이터는 이벤트참여 릴레이션에 삽입할 수 없음
- 삽입하려면 실제로 참여하지 않은 임시 이벤트번호를 함께 삽입해야 함

고객아이디	이벤트번호	당첨여부	고객이름	등급
apple	E001	Y	정소화	gold
apple	E005	N	정소화	gold
apple	E010	Y	정소화	gold
banana	E002	N	김선우	vip
banana	E005	Y	김선우	vip
carrot	E003	Y	고명석	gold
carrot	E007	Y	고명석	gold
orange	E004	N	김용욱	silver
melon	NULL	NULL	성원용	gold

← 삽입 불가!



✓ 갱신 이상(update anomaly)

○ 릴레이션의 중복된 튜플들 중 일부만 수정하여 데이터가 불일치하게 되는 모순이 발생하는 문제

- 아이디가 “apple”인 고객의 등급이 “gold”에서 “vip”로 변경되었는데, 일부 튜플에 대해서만 등급이 수정된다면 “apple” 고객이 서로 다른 등급을 가지는 모순이 발생

고객아이디	이벤트번호	당첨여부	고객이름	등급
apple	E001	Y	정소화	vip
apple	E005	N	정소화	vip
apple	E010	Y	정소화	gold
banana	E002	N	김선우	vip
banana	E005	Y	김선우	vip
carrot	E003	Y	고명석	gold
carrot	E007	Y	고명석	gold
orange	E004	N	김용욱	silver

← 데이터 불일치 발생!



✓ 삭제 이상(deletion anomaly)

- 릴레이션에서 튜플을 삭제하면 꼭 필요한 데이터까지 손실되는 연쇄 삭제 현상이 발생하는 문제
 - 아이디가 “orange”인 고객이 이벤트 참여를 취소해 관련 튜플을 삭제하게 되면 이벤트 참여와 관련이 없는 고객아이디, 고객이름, 등급 데이터까지 손실됨

<u>고객아이디</u>	<u>이벤트번호</u>	당첨여부	고객이름	등급
apple	E001	Y	정소화	gold
apple	E005	N	정소화	gold
apple	E010	Y	정소화	gold
banana	E002	N	김선우	vip
banana	E005	Y	김선우	vip
carrot	E003	Y	고명석	gold
carrot	E007	Y	고명석	gold
orange	E004	N	김용욱	silver

← 데이터 손실 발생!

✓ 정규화

- 이상 현상이 발생하지 않도록, 릴레이션을 관련 있는 속성들로만 구성하기 위해 릴레이션을 분해 (decomposition)하는 과정
- 함수적 종속성을 판단하여 정규화를 수행함

✓ 함수적 종속성(FD; Functional Dependency)

- 속성들 간의 관련성

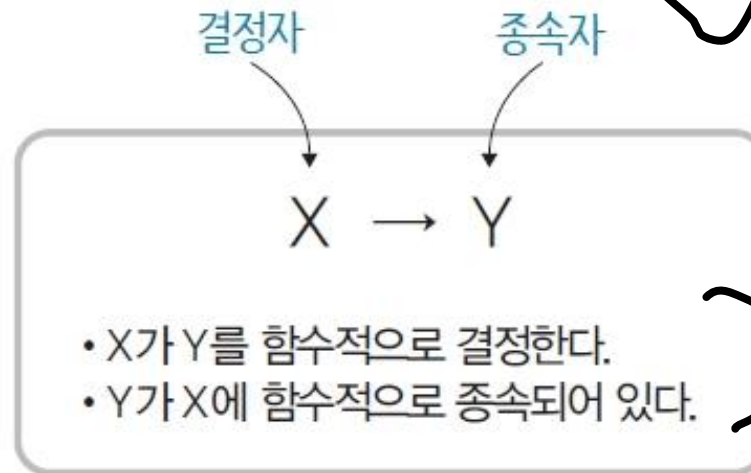
정규화

함수적 종속성을 이용해,
릴레이션을 연관성이 있는 속성들로만 구성되도록
분해하여 이상 현상이 발생하지 않는
올바른 릴레이션으로 만들어 나가는 과정

✓ 함수 종속

○ “X가 Y를 함수적으로 결정한다” ✓

- 릴레이션 내의 모든 튜플에서 하나의 X 값에 대한 Y 값이 항상 하나임
- X와 Y는 하나의 릴레이션을 구성하는 속성들의 부분 집합
- “Y가 X에 함수적으로 종속되어 있다”와 같은 의미
- $X \rightarrow Y$ 로 표현(X는 결정자, Y는 종속자)



✓ 함수 종속 관계 판단 예 (1)

○ 고객 릴레이션

고객아이디	고객이름	등급
apple	정소화	gold
banana	김선우	vip
carrot	고명석	gold
orange	김용욱	silver

각 고객아이디 속성 값에 대응되는
고객 이름과 등급 속성 값이
단 하나임

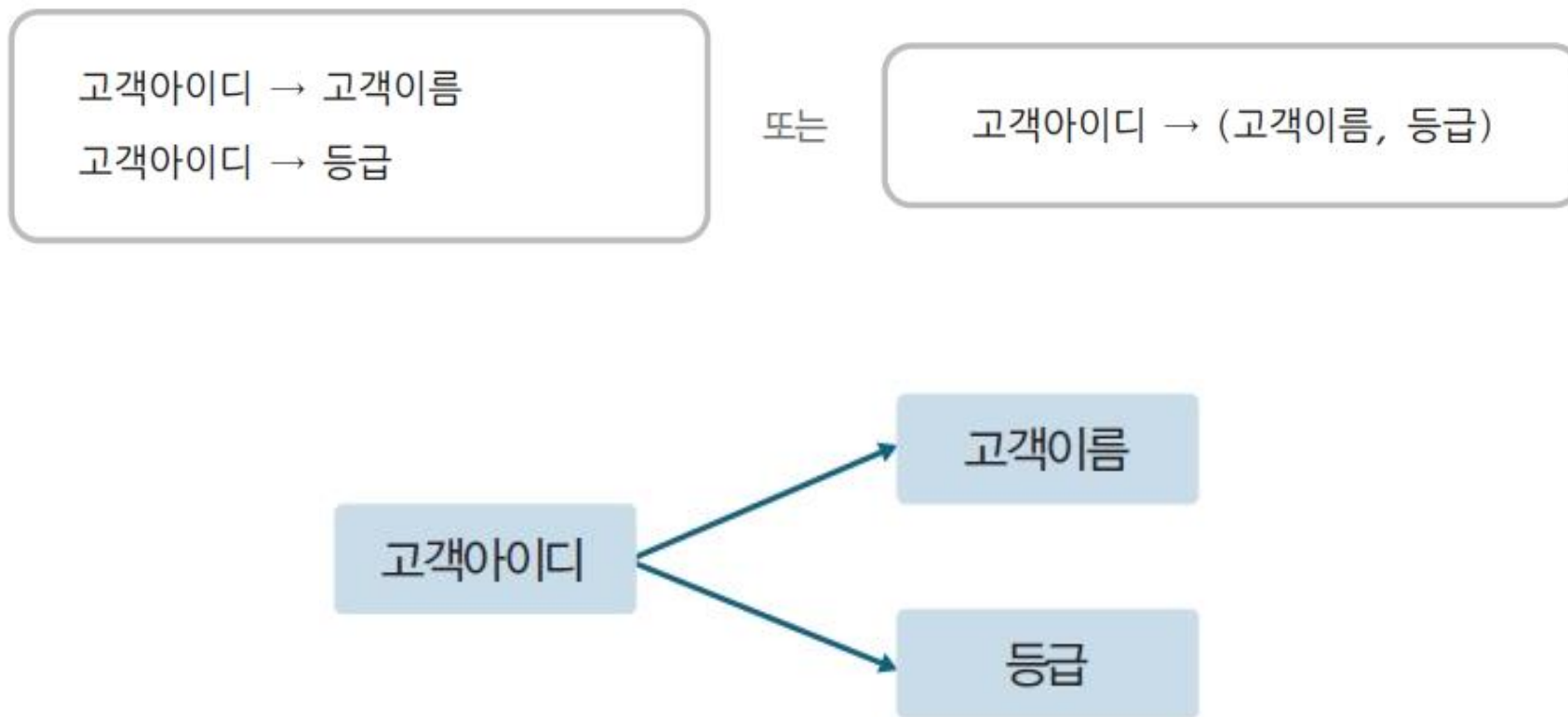
고객아이디 → 고객이름
고객아이디 → 등급

또는

고객아이디 → (고객이름, 등급)

✓ 함수 종속 관계 판단 예 (1)

- 함수 종속 다이어그램 : 함수 종속 관계를 도식화하여 표현한 것



✓ 함수 종속 관계 판단 시 유의 사항

- 속성 자체의 특성과 의미를 기반으로 함수 종속성을 판단해야 함
 - 속성 값은 계속 변할 수 있으므로 현재 릴레이션에 포함된 속성 값만으로 판단하면 안됨
- 일반적으로 기본키와 후보키는 릴레이션의 다른 모든 속성들을 함수적으로 결정함
- 기본키나 후보키가 아니어도 다른 속성 값을 유일하게 결정하는 속성은 함수 종속 관계에서 결정자가 될 수 있음

✓ 함수 종속 관계 판단 예 (2)

○ 이벤트 참여 릴레이션

<u>고객아이디</u>	<u>이벤트번호</u>	당첨여부	고객이름
apple	E001	Y	정소화
apple	E005	N	정소화
apple	E010	Y	정소화
banana	E002	N	김선우
banana	E005	Y	김선우
carrot	E003	Y	고명석
carrot	E007	Y	고명석
orange	E004	N	김용욱

✓ 함수 종속 관계 판단 예 (2)

○ 이벤트 참여 릴레이션에 존재하는 함수 종속 관계

고객아이디 → 고객이름

{고객아이디, 이벤트번호} → 당첨여부

{고객아이디, 이벤트번호} → 고객이름

고객이름은 {고객아이디, 이벤트번호}의
일부분인 고객아이디에 종속되어 있음

→ **고객이름은 {고객아이디, 이벤트 번호}에
부분 함수 종속됨**



✓ 완전 함수 종속(FFD; Full Functional Dependency)

- 릴레이션에서 속성 집합 Y가 속성 집합 X에 함수적으로 종속되어 있지만, 속성 집합 X의 전체가 아닌 일부분에는 종속되지 않음을 의미
- 일반적으로 함수 종속은 완전 함수 종속을 의미함
- 예) 당첨여부는 {고객아이디, 이벤트번호}에 완전 함수 종속됨

✓ 부분 함수 종속(PFD; Partial Functional Dependency)

- 릴레이션에서 속성 집합 Y가 속성 집합 X의 전체가 아닌 일부분에도 함수적으로 종속됨을 의미
- 예) 고객이름은 {고객아이디, 이벤트번호}에 부분 함수 종속됨

✓ 고려할 필요가 없는 함수 종속 관계

- 결정자와 종속자가 같거나, 결정자가 종속자를 포함하는 것처럼 당연한 함수 종속 관계는 고려하지 않음

고객아이디 → 고객아이디

{고객아이디, 이벤트번호} → 이벤트번호

✓ 정규화(normalization)

○ 개념

- 함수 종속성을 이용해 릴레이션을 연관성이 있는 속성들로만 구성되도록 분해해서, 이상 현상이 발생하지 않는 올바른 릴레이션으로 만들어 가는 과정
 - 기본 목표 : 관련이 없는 함수 종속성은 별개의 릴레이션으로 표현

○ 주의 사항

- 정규화를 통해 릴레이션은 무손실 분해(nonloss decomposition)되어야 함
- 릴레이션이 의미상 동등한 릴레이션들로 분해되어야 하고, 분해로 인한 정보 손실이 발생하지 않아야 함
- 분해된 릴레이션들을 자연 조인하면 분해 전의 릴레이션으로 복원 가능해야 함

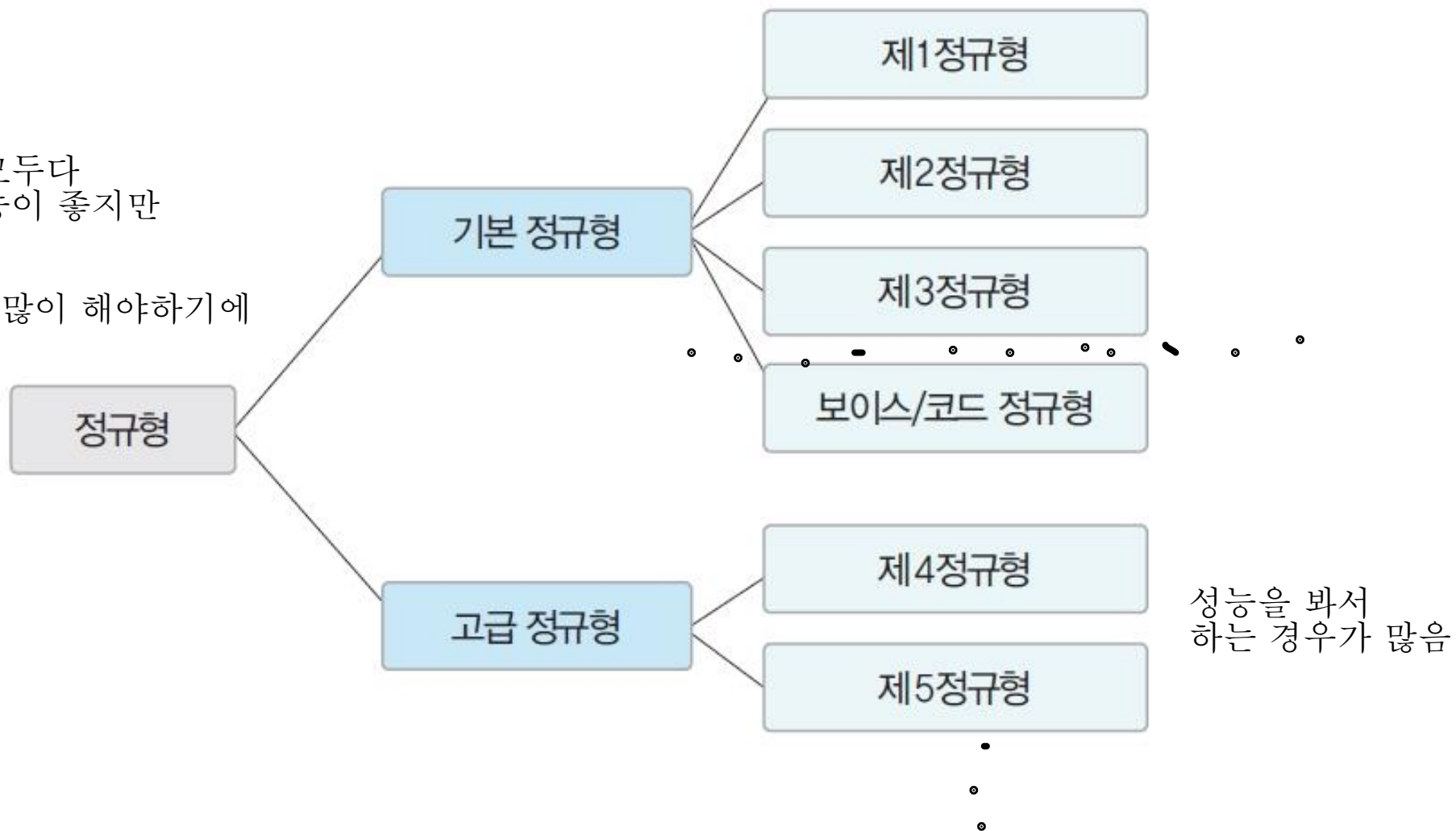


✓ 정규형(NF; Normal Form)

- 릴레이션이 정규화된 정도
- 각 정규형마다 제약조건이 존재
 - 정규형의 차수가 높아질수록 요구되는 제약조건이 많아지고 엄격해짐
- 릴레이션의 특성을 고려해서 적합한 정규형을 선택

정규형의 종류

이론적으로는 모두다
만족하는게 성능이 좋지만
실전에서는
조인을 또 다시 많이 해야하기에



✓ 정규형들의 관계



✓ 제1정규형(1NF; First Normal Form)

배열처리 안되게

- 릴레이션의 모든 속성이 더는 분해되지 않는 원자 값(atomic value)만 가지면 제1정규형을 만족함
- 제1정규형을 만족해야 관계 데이터베이스의 릴레이션이 될 자격이 있음

제1정규형(1NF)

릴레이션에 속한 모든 속성의 도메인이 원자 값(atomic value)으로만 구성되어 있으면 제1 정규형에 속한다.

제1정규형을 만족하지 않는 릴레이션

고객아이디	이벤트번호	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%

배열처리되어있음

✓ 제1정규형(1NF; First Normal Form)

제1정규형을 만족하는 릴레이션

<u>고객아이디</u>	<u>이벤트번호</u>	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%

✓ 제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

제1정규형을 만족하지만 데이터 중복으로 인한 이상 현상이 발생함

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%

종속 관계 따져보자!

✓ 제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 이벤트 참여 릴레이션에 존재하는 함수 종속 관계

고객아이디 → 등급

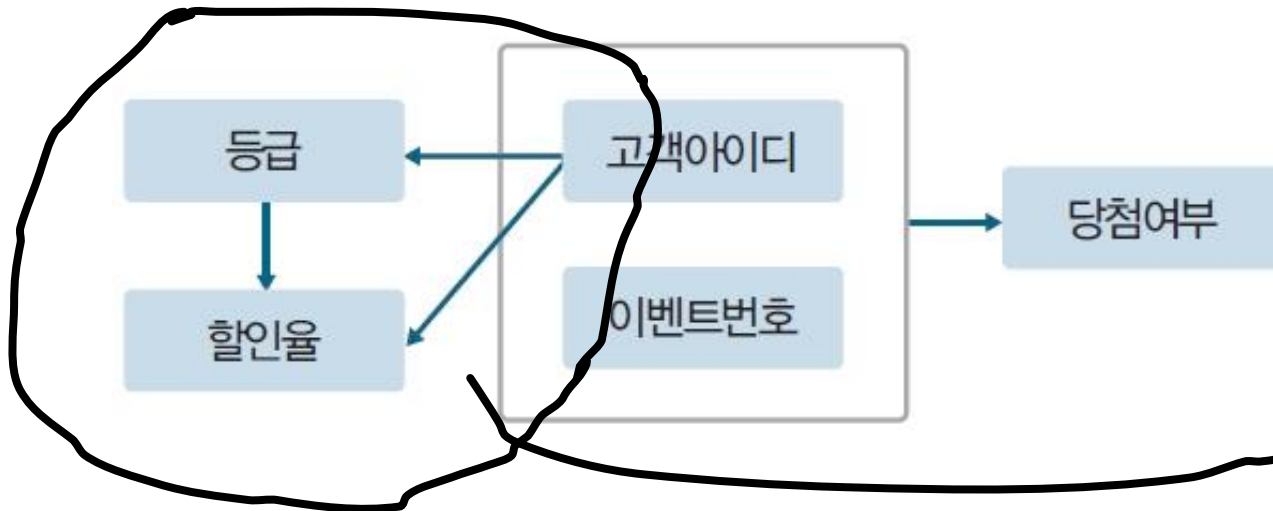
고객아이디 → 할인율

등급 → 할인율

{고객아이디, 이벤트번호} → 당첨여부

고객 아이디로 인해서
등급과 할인율이 정해진다

고객 아이디와 이벤트 번호를 따져서 당첨 여부가 결정된다



이 모양이 나오면 2,3정규형 이상(만족하지 못함)이다

✓ 제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 삽입 이상

<u>고객아이디</u>	<u>이벤트번호</u>	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%
grape	NULL	NULL	silver	5%

← 삽입 불가!

✓ 제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 갱신 이상

고객아이디	이벤트번호	당첨여부	등급	할인율
apple	E001	Y	vip	10%
apple	E005	N	vip	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%

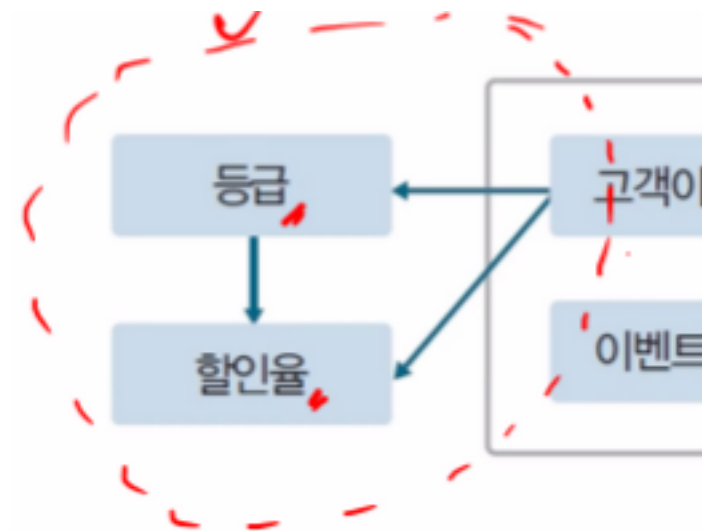
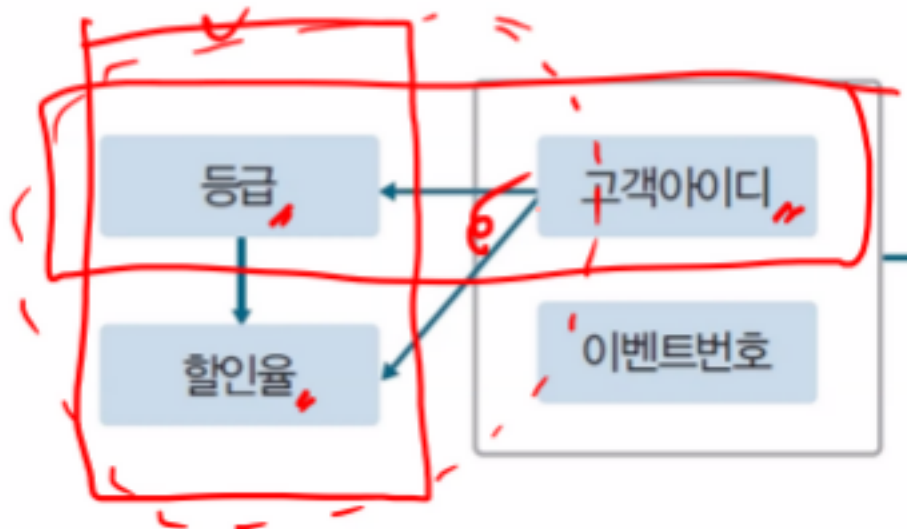
← 데이터 불일치 발생!

제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

- 삭제 이상: 회원 자체가 날라감. 이번 이벤트를 안한다는 뜻인디

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%
		Y	vip	20%
		Y	gold	10%
		Y	gold	10%
		N	silver	5%

이렇게 바뀌야 함
결과적으로 2개로 분리



이모양때문에 그럼
고객 아이디 -> 등급
고객 아이디 -> 할인율
등급 -> 할인율

데이터 손실 발생!

고객아이디 -> 등급 -> 할인율 모양으로 되게.

제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

이상 현상의 발생 이유는? **기본키에 완전 함수 종속되지 못한 등급과 할인율 때문**

고객아이디, 이벤트번호 -> 등급
 고객아이디, 이벤트번호 -> 할인율
 이 아니라서 문제가 된다고! 이게 이상현상의발생이유라고

기본키가
 고객아이디, 이벤트번호
 조합임!!!!!!!!!!

고객아이디	이벤트번호	당첨여부	등급	할인율
apple	E001	Y	vip	10%
apple	E005	N	vip	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%
grape	NULL	NULL	silver	5%

고객아이디, 이벤트번호 -> 당첨여부
 고객아이디 -> 등급
 고객아이디 -> 할인율
 등급 -> 할인율

← 데이터 불일치로 인한 갱신 이상

← 데이터 손실로 인한 삭제 이상

← 삽입 불가로 인한 삽입 이상

✓ 제1정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 이상 현상의 발생 이유

- 기본키인 {고객아이디, 이벤트번호}에 완전 함수 종속되지 못하고 일부분인 고객아이디에 종속되는 등급과 할인율 속성이 존재하기 때문

○ 문제 해결 방법

- 부분 함수 종속이 제거되도록 이벤트참여 릴레이션을 분해
→ 분해된 릴레이션들은 제2정규형에 속하게 됨

고객아이디, 이벤트번호 → 등급
고객아이디, 이벤트번호 → 할인율

=>

둘다 고객 아이디에만 함수종속적이니

고객아이디 등급 할인율 칼럼을 가진 테이블이 2개
따로 만들어 분리

✓ 제2정규형(2NF; Second Normal Form)

- 릴레이션이 제1정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 완전 함수 종속되면 제2정규형을 만족함
- 제1정규형에 속하는 릴레이션이 제2정규형을 만족하게 하려면?
 - 부분 함수 종속을 제거하고 모든 속성이 기본키에 완전 함수 종속되도록 분해

제2정규형(2NF)

릴레이션이 제1정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 완전 함수 종속되면 제2정규형에 속한다.

✓ 제2정규형(2NF; Second Normal Form)

제1정규형을 만족하지만 제2정규형은 만족하지 않는 릴레이션
(기본키에 완전 함수 종속되지 않은 등급과 할인율 속성 때문)

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%

✓ 제2정규형(2NF; Second Normal Form)

○ 제2정규형을 만족하도록 분해된 2개의 릴레이션

분해 전의 이벤트참여 릴레이션

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%

고객 릴레이션과 이벤트참여 릴레이션은
모두 제2정규형에 속함

부분 함수 종속을 제거하려고 분해

고객 릴레이션

고객아이디	등급	할인율
apple	gold	10%
banana	vip	20%
carrot	gold	10%
orange	silver	5%

고객아이디
->
등급
->
할인율

하지만 이 테이블이
또 문제다. 제3정규형
만족X

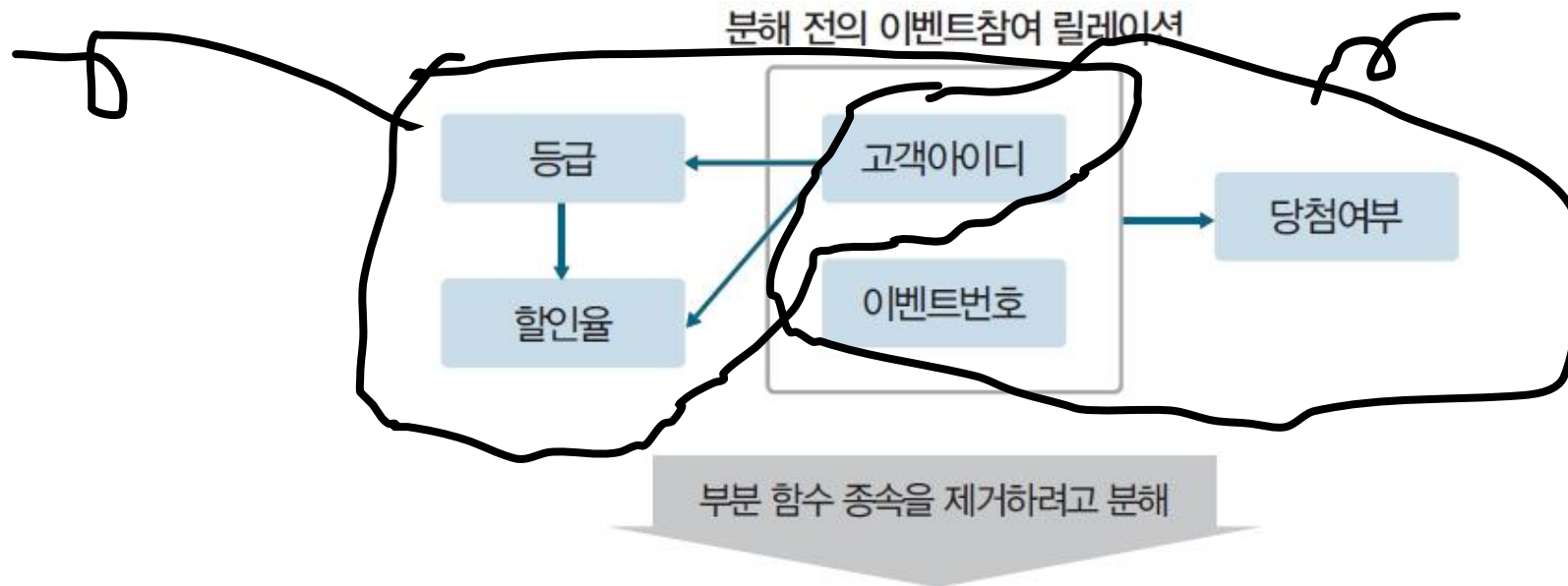
이벤트참여 릴레이션

고객아이디	이벤트번호	당첨여부
apple	E001	Y
apple	E005	N
apple	E010	Y
banana	E002	N
banana	E005	Y
carrot	E003	Y
carrot	E007	Y
orange	E004	N

고객 아이디 + 이벤트 번호
=>
당첨 여부 결정

제2정규형(2NF; Second Normal Form)

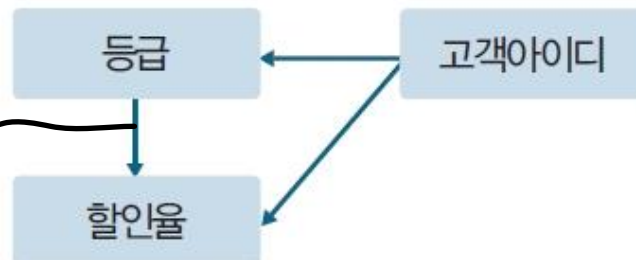
고객 릴레이션과 이벤트 참여 릴레이션의 함수 종속 다이어그램



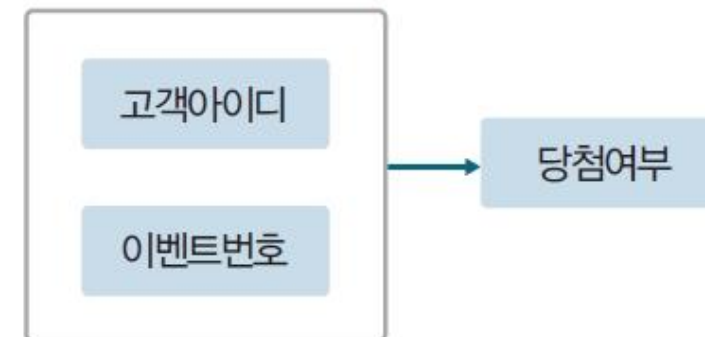
당첨 여부와
등급+할인율은
상관없기에

전 테이블을
표현한
다이어그램

고객 릴레이션



이벤트참여 릴레이션



제2정규형(2NF; Second Normal Form)

이벤트 참여 릴레이션의 분해 결과

분해 전의 이벤트참여 릴레이션

부분 함수 종속을 제거하려고 분해

고객 릴레이션

고객아이디	등급	할인율
apple	gold	10%
banana	vip	20%
carrot	gold	10%
orange	silver	5%

함수 종속성을 여러 개 포함하여
이상 현상이 발생할 수 있음

둘이 달라
질수있음

이벤트참여 릴레이션

고객아이디	이벤트번호	당첨여부
apple	E001	Y
apple	E005	N
apple	E010	Y
banana	E002	N
banana	E005	Y
carrot	E003	Y
carrot	E007	Y
orange	E004	N

함수 종속성을 하나만 포함하여 이상 현상 발생하지 않음

부분 함수 종속 해소

✓ 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

제2정규형을 만족하지만 이상 현상이 발생함

고객 릴레이션

<u>고객아이디</u>	등급	할인율
apple	gold	10%
banana	vip	20%
carrot	gold	10%
orange	silver	5%

✓ 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 삽입 이상

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

← 삽입 불가! ✓

등급에 따른 할인율을 추가하고 싶은데

✓ 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 갱신 이상

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

데이터 불일치 발생!



✓ 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 삭제 이상

고객아이디	등급	할인율
apple	gold	10%
banana ✓	vip	20%
carrot	gold	10%
orange	silver	5%

← 데이터 손실 발생! ✓

회원정보만 삭제인디
등급당 할인율 정보도 삭제...

- 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

이상 현상의 발생 이유는? 이행적 함수 종속이 존재하기 때문

이로함!!!
= =

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

✓ 제2정규형은 만족하지만 이상 현상이 발생하는 릴레이션 예

○ 이상 현상의 발생 이유

- 이행적 함수 종속이 존재하기 때문

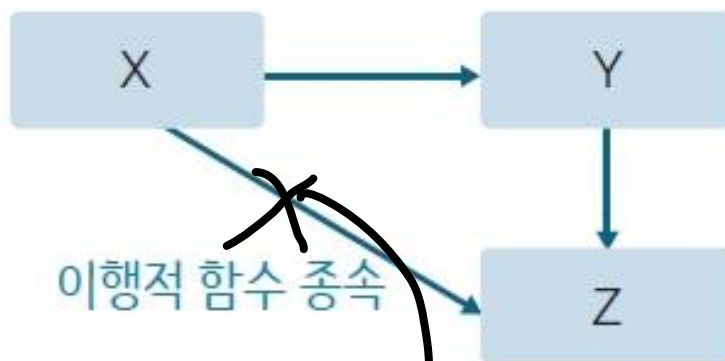
○ 문제 해결 방법

- 이행적 함수 종속이 제거되도록 고객 릴레이션을 분해
→ 분해된 릴레이션들은 제3정규형에 속하게 됨

해결은 분리!

이행적 함수 종속(transitive FD)

- 릴레이션을 구성하는 3개의 속성 집합 X, Y, Z에 대해 함수 종속 관계 $X \rightarrow Y$ 와 $Y \rightarrow Z$ 가 존재하면 논리적으로 $X \rightarrow Z$ 가 성립되는데, 논리적으로 맞긴 맞는데 이상현상을 야기할 수 있는 이행적 함수종속!
- 이때 Z가 X에 이행적으로 함수 종속되었다고 함



$x \rightarrow y$

$y \rightarrow z$

이렇게

테이블을 분리하므로써
 $x \rightarrow z$ 를 꺼야 함

✓ 제3정규형(3NF; Third Normal Form)

- 릴레이션이 제2정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 이행적 함수 종속이 되지 않으면 제3정규형을 만족함
- 제2정규형에 속하는 릴레이션이 제3정규형을 만족하게 하려면?
 - 모든 속성이 기본키에 이행적 함수 종속이 되지 않도록 분해

제3정규형(3NF)

릴레이션이 제2정규형에 속하고, 기본키가 아닌 모든 속성이 기본키에 이행적 함수 종속이 되지 않으면 제3정규형에 속한다.

✓ 제3정규형(3NF; Third Normal Form)

제2정규형을 만족하지만 제3정규형은 만족하지 않는 릴레이션
(고객아이디가 등급을 통해 할인율을 결정하는
이행적 함수 종속 관계가 존재하기 때문)

<u>고객아이디</u>	등급	할인율
apple	gold	10%
banana	vip	20%
carrot	gold	10%
orange	silver	5%

✓ 제3정규형(3NF; Third Normal Form)

○ 제3정규형을 만족하도록 분해된 2개의 릴레이션

분해 전의 고객 릴레이션

고객아이디	등급	할인율
apple	gold	10%
banana	vip	20%
carrot	gold	10%
orange	silver	5%

이행적 함수 종속을 제거하려고 분해

고객 릴레이션과 고객등급 릴레이션은
모두 제3정규형에 속함

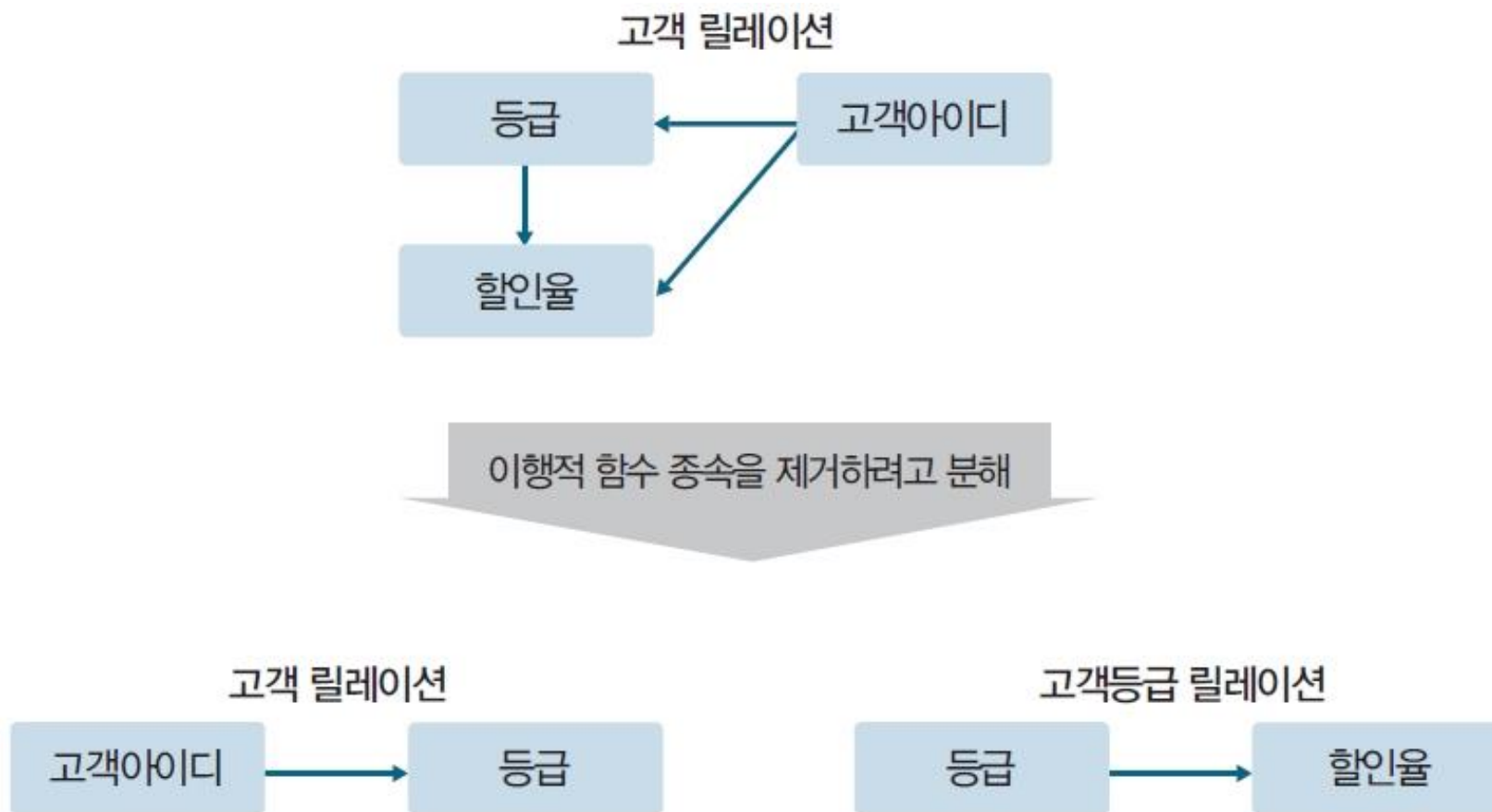
고객 릴레이션

고객아이디	등급
apple	gold
banana	vip
carrot	gold
orange	silver

고객등급 릴레이션

등급	할인율
gold	10%
vip	20%
silver	5%

✓ 제3정규형(3NF; Third Normal Form)



✓ 보이스/코드 정규형(BCNF; Boyce/Codd Normal Form) 3.5

○ 필요성

- 하나의 릴레이션에 여러 개의 후보키가 존재하는 경우, 제3정규형까지 모두 만족해도 이상 현상이 발생할 수 있음

○ 의미

- 강한 제3정규형(strong 3NF)
 - 후보키를 여러 개 가지고 있는 릴레이션에 발생할 수 있는 이상 현상을 해결하기 위해 제3정규형보다 좀 더 엄격한 제약조건을 제시
 - 보이스/코드 정규형에 속하는 모든 릴레이션은 제3정규형에 속하지만, 제3정규형에 속하는 모든 릴레이션이 보이스/코드 정규형에 속하는 것은 아님

제3정규형) BCNF

보이스/코드 정규형(BCNF)

릴레이션의 함수 종속 관계에서 모든 결정자가 후보키이면 보이스/코드 정규형에 속한다.

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예

○ 강좌신청 릴레이션

기본키는 아니지만
기본키가 될 자격이 있어서
고객아이디, 담당강사번호 -> 인터넷강좌
로 표현할 수 있음.
기본키에 대한 재귀적 형태를 뚫수있음.

기본키의 일부분 + 다른속성 -> 기본키의 다른 일부분 될 수 있음

제3정규형을 만족하지만 보이스/코드 정규형은 만족하지 않는 릴레이션
(함수 종속 관계에서 모든 결정자가 후보키가 아니기 때문)

고객아이디	인터넷강좌	담당강사번호
apple	영어회화	P001
banana	기초토익	P002
carrot	영어회화	P001
carrot	기초토익	P004
orange	영어회화	P003
orange	기초토익	P004

[강좌신청 릴레이션의 후보키]

{고객아이디, 인터넷강좌} : 기본키

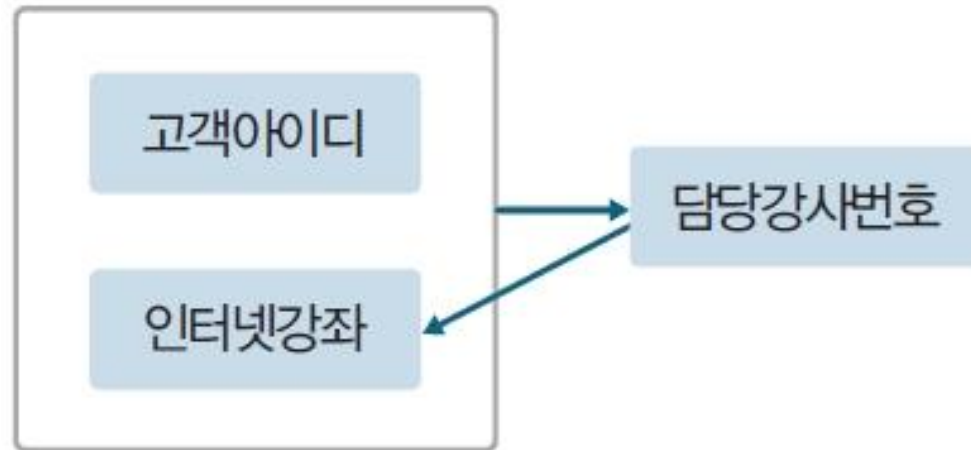
{고객아이디, 담당강사번호}

고객아이디 + 인터넷강좌 조합

담당강사번호 + 고객아이디로 조합

으로 각 행들을 모두 분리시킬수 있을때(기본키일때)
BCNF 만족X

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예



이것도 분리로 해결 가능

고객아이디+인터넷강좌
-> 담당강사번호

담당강사번호->인터넷강좌

이런형태는 직관적이지 않음.

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예

○ 삽입 이상

고객아이디	인터넷강좌	담당강사번호
apple	영어회화	P001
banana	기초토익	P002
carrot	영어회화	P001
carrot	기초토익	P004
orange	영어회화	P003
orange	기초토익	P004
NULL	중급토익	P005

← 삽입 불가!

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예

○ 갱신 이상

고객아이디	인터넷강좌	담당강사번호
apple	영어회화	P001
banana	기초토익	P002
carrot	영어회화	P001
carrot	중급토익	P004
orange	영어회화	P003
orange	기초토익	P004

데이터 불일치 발생!

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예

○ 삭제 이상

<u>고객아이디</u>	<u>인터넷강좌</u>	담당강사번호
apple	영어회화	P001
banana	기초토익	P002
carrot	영어회화	P001
carrot	기초토익	P004
orange	영어회화	P003
orange	기초토익	P004

← 데이터 손실 발생!

✓ 보이스/코드 정규형을 만족하지 않는 릴레이션 예

이상 현상의 발생 이유는?
 담당강사번호 하나만이 후보키가 아님에도
담당강사번호가 후보키가 아님에도 인터넷강좌 속성을 결정하기 때문



고객아이디	인터넷강좌	담당강사번호	
apple	영어회화	P001	
banana	기초토익	P002	← 데이터 손실로 인한 삭제 이상
carrot	영어회화	P001	
carrot	중급토익	P004	← 데이터 불일치로 인한 갱신 이상
orange	영어회화	P003	
orange	기초토익	P004	
NULL	중급토익	P005	← 삽입 불가로 인한 삽입 이상

✓ 보이스/코드 정규형(BCNF; Boyce/Codd Normal Form)

○ BCNF를 만족하도록 분해된 2개의 릴레이션

강좌신청 릴레이션

고객아이디	인터넷강좌	담당강사번호
apple	영어회화	P001
banana	기초토익	P002
carrot	영어회화	P001
carrot	기초토익	P004
orange	영어회화	P003
orange	기초토익	P004

후보키가 아닌 결정자를
제거하려고 분해

고객담당강사 릴레이션과
강좌담당 릴레이션은
모두 BCNF에 속함

고객담당강사 릴레이션

고객아이디	담당강사번호
apple	P001
banana	P002
carrot	P001
carrot	P004
orange	P003
orange	P004

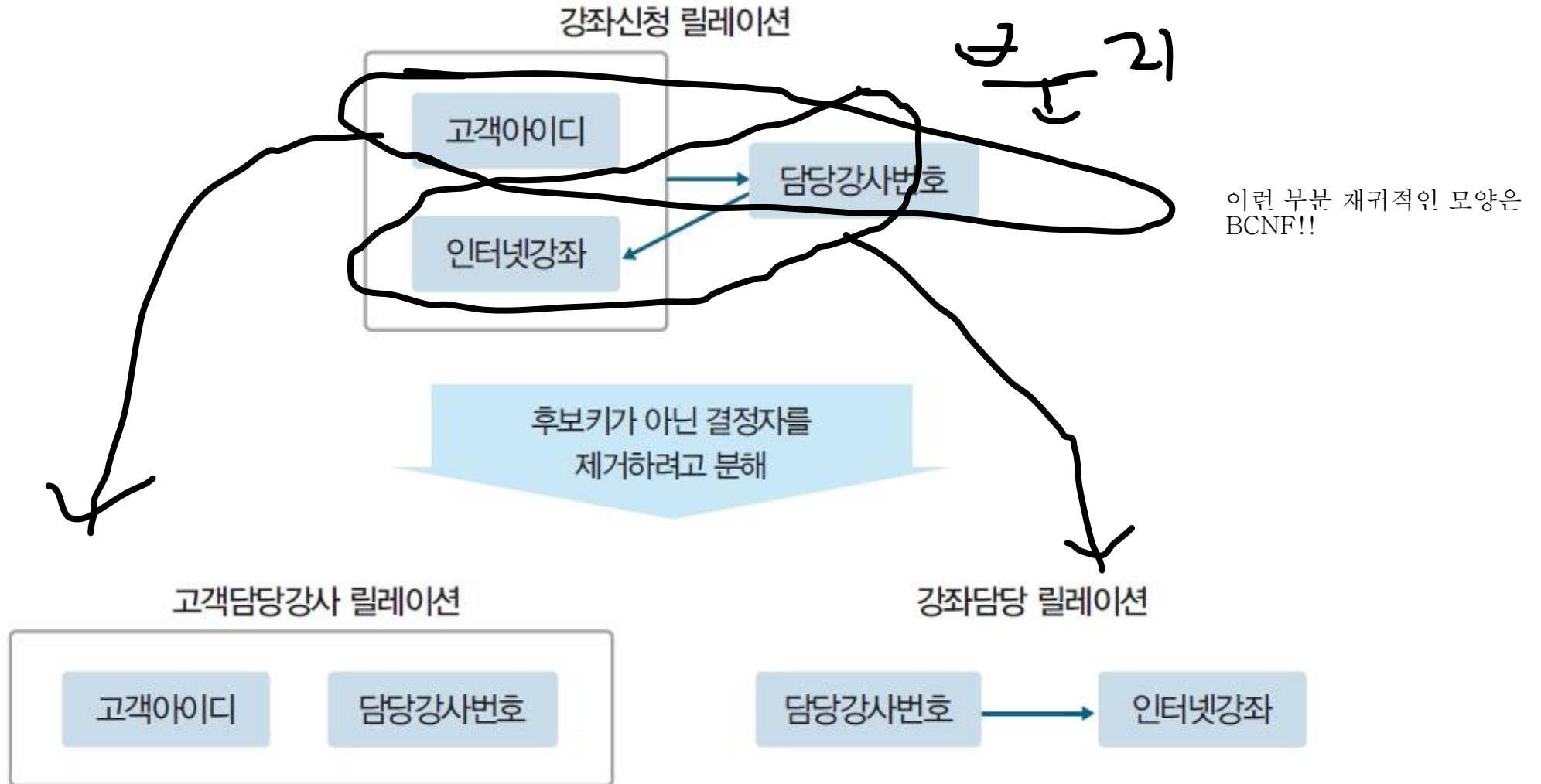
강좌담당 릴레이션

담당강사번호	인터넷강좌
P001	영어회화
P002	기초토익
P003	영어회화
P004	기초토익

↙ ↘

✓ 보이스/코드 정규형(BCNF; Boyce/Codd Normal Form)

○ 고객담당강사 릴레이션과 강좌담당 릴레이션의 함수 종속 다이어그램



✓ 제4정규형

1정규형과 비슷.

한 속성에 여러가지 들어가는 것
1정규형과 다른 점은 속성관 연결점이 없음

- 릴레이션이 보이스/코드 정규형을 만족하면서, 함수 종속이 아닌 다치 종속(MVD; Multi Valued Dependency)을 제거하면 제4정규형에 속함

✓ 제5정규형

- 릴레이션이 제4정규형을 만족하면서, 후보키를 통하지 않는 조인 종속 (JD; Join Dependency)을 제거하면 제5정규형에 속함

✓ 정규화 시 주의 사항

- 모든 릴레이션이 제5정규형에 속해야만 바람직한 것은 아님
- 일반적으로 제3정규형이나 보이스/코드 정규형에 속하도록 릴레이션을 분해하여 데이터 중복을 줄이고 이상 현상을 해결하는 경우가 많음

✓ 정규화 과정

정규화를 진행하며
너무 잘게 쪼개지면
성능에 오히려 악영향
조인이 많이 필요해지므로

실상은 테이블 설계시
점진적인 방법으로 정규화
진행이 잘 안된다.



정규화는
차례대로 하는 것이
이론적이긴해

BCNF까지 하려고 해라