

데이터 베이스 정규화

- 정규화(Normalization)

- 이상현상이 있는 릴레이션(테이블)을 분해하여 이상현상을 없애는 과정
- 이상현상이 존재하는 릴레이션을 분해하여 여러 개의 릴레이션을 생성
- 정규형이 높아질 수록 이상현상은 줄어든다.
- 정규화는 전 단계가 이루어져 있어야 다음 단계로 넘어갈 수 있으며 아래의 그림과 같이 단계가 있다.



데이터 베이스 정규화

- 이상현상 (Anomaly)

- 삽입 이상 : 자료를 삽입할 때 의도하지 않은 자료까지 삽입해야만 자료 삽입이 가능하다
- 갱신 이상 : 중복된 데이터 중 일부만 수정되어 데이터 모순이 일어나는 현상
- 삭제 이상 : 어떤 정보를 삭제하면, 의도하지 않은 다른 정보까지 연쇄되어 삭제되는 현상

데이터 베이스 정규화

학번	이름	나이	성별	강의코드	강의명	전화번호
1011	이태호	23	남	AC1	데이터베이스 개론	010-2627-8123
1012	강민정	20	여	AC2	운영체제	010-4665-1941
1013	김현수	21	남	AC3	자료구조	010-5223-4464
1013	김현수	21	남	AC4	웹 프로그래밍	010-5223-4464
1014	이병철	26	남	AC5	알고리즘	010-6305-2912

옆의 예제로 봤을 때 이상 현상들

1. 삽입 이상

= 강의를 수강하지 않은 새로운 학생을 삽입할 경우
강의명 속성에 NULL 이 들어가야 한다.

2. 갱신 이상

= 강의 코드가 AC3인 김현수의 전화번호를 수정할 경우
3번째만 수정되고 4번째는 수정되지 않을 것이다.

3. 삭제 이상

= 강의 코드 AC1만 삭제할 경우
이태호 학생 데이터도 삭제될 것이다.

데이터 베이스 정규화

- 정규화의 장단점

= 장점

이상현상 제거

데이터베이스를 수정할 시, 구조를 변경하지 않아도 된다.

연동된 응용 프로그램에 최소한의 영향을 미침

= 단점

JOIN 연산이 많아진다.

응답 시간이 길어질 수 있다.

반정규화를 적용할 수 있다.

- 정규화의 목적

- 불필요한 데이터를 제거해 중복을 최소화
- 삽입/갱신/삭제 시 발생할 수 있는 각종 이상현상을 방지

데이터 베이스 정규화

• 함수 종속

“ X 가 Y 를 함수적으로 결정한다 ”

- 릴레이션 내의 모든 튜플에서 하나의 X 값에 대한 Y 값이 항상 하나이다.
- X 와 Y 는 하나의 릴레이션을 구성하는 속성들의 부분 집합이다.
- “ Y 가 X 에 함수적으로 종속되어 있다 ” 와 같은 의미
- $X \rightarrow Y$ 라고 표현한다. (X 는 결정자, Y 는 종속자)

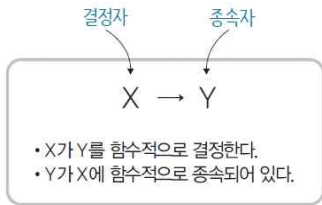


그림 9-6 함수 종속의 표현

데이터 베이스 정규화

• 함수 종속 관계 판단 예 (1)

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

각 고객아이디 속성 값에 대응되는
고객이름과 등급 속성 값이 단 하나임

그림 9-7 함수 종속 관계 설명을 위한 릴레이션의 예 : 고객 릴레이션

고객아이디 → 고객이름

고객아이디 → 등급

또는

고객아이디 → (고객이름, 등급)

그림 9-8 고객 릴레이션에 존재하는 함수 종속 관계

데이터 베이스 정규화

- 함수 종속 관계 판단 예 (1)



그림 9-8 고객 릴레이션에 존재하는 함수 종속 관계

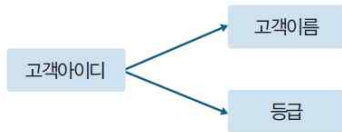


그림 9-9 고객 릴레이션의 함수 종속 다이어그램

데이터 베이스 정규화

- 함수 종속 관계 판단 시 유의 사항

- 속성 자체의 특성과 의미를 기반으로 판단해야 함.
 - = 속성 값은 계속 변할 수 있기 때문에 다른 경우까지 생각해야 함
- 일반적으로 기본키와 후보키는 릴레이션의 다른 모든 속성들을 함수적으로 결정함.
- 기본키나 후보키가 아니어도 다른 속성 값을 유일하게 결정하는 속성은 함수 종속 관계에서 결정자가 될 수 있다.

데이터 베이스 정규화

• 함수 종속 관계 판단 예 (2)

고객아이디	이벤트번호	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

그림 9-10 함수 종속 관계를 설명하기 위한 릴레이션의 예 : 이벤트참여 릴레이션



그림 9-12 이벤트참여 릴레이션의 함수 종속 다이어그램

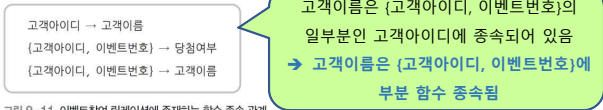


그림 9-11 이벤트참여 릴레이션에 존재하는 함수 종속 관계

데이터 베이스 정규화

- 부분 함수 종속

- 릴레이션에서 속성 집합 Y 가 속성 집합 X 의 전체가 아닌 일부분에도 함수적으로 종속됨을 의미
- 예) 고객이름은 { 고객아이디, 이벤트번호 } 에 부분 함수 종속이다.

- 완전 함수 종속

- 릴레이션에서 속성 집합 Y 가 속성 집합 X 에 함수적으로 종속되어 있지만, 속성 집합 X 의 전체가 아닌 일부분에는 종속되지 않음을 의미
- 일반적으로 함수 종속은 완전 함수 종속을 의미함
- 예) 당첨여부는 { 고객아이디, 이벤트번호 } 에 완전 함수 종속이다.

데이터 베이스 정규화

- 고려할 필요가 없는 함수 종속 관계

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

고객아이디 \rightarrow 고객아이디

{고객아이디, 이벤트번호} \rightarrow 이벤트번호

그림 9-13 고려할 필요가 없는 함수 종속 관계의 예

데이터 베이스 정규화

- 제 1 정규화 (1NF)

= 정의

테이블의 컬럼은 원자 값(하나의 값)을 갖도록 테이블을 분해한다.

1. 각 컬럼이 하나의 속성만을 가져야 한다.
2. 하나의 컬럼은 같은 종류나 타입의 값을 가져야 한다.
3. 각 컬럼이 유일한 이름을 가져야 한다.
4. 컬럼의 순서가 상관 없어야 한다.

이름	나이	수강과목
홍길동	20	C, C++
이순신	21	Java
이 산	22	DB, 운영체제



이름	나이	수강과목
홍길동	20	C
홍길동	20	C++
이순신	21	Java
이 산	22	DB
이 산	22	운영체제

데이터 베이스 정규화

- 제 2 정규화 (2 NF)


= 정의

부분 함수적 종속을 제거해서 완전 함수 종속을 만든다.

- 완전 함수 종속이란 ?

: 기본키의 부분집합이 결정자가 되어서는 안된다.

학생번호	과목	지도교수	성적
101	운영체제	김운체	100
101	DB	조디비	60
102	자바	박자바	70
103	C	김씨	80
103	C++	이씨플	90



학생번호	과목	성적
101	운영체제	100
101	DB	60
102	자바	70
103	C	80
103	C++	90

과목	지도교수
운영체제	김운체
DB	조디비
자바	박자바
C	김씨
C++	이씨플

데이터 베이스 정규화

- 제 3 정규화 (3 NF)


= 정의

테이블에 대해 이행적 종속을 없애도록 테이블을 분해

- 이행적 종속이란 ?

: $X \rightarrow Y$ 이고, $Y \rightarrow Z$ 일 때, $X \rightarrow Z$ 가 성립하는 것

학생번호	강좌이름	수강료
501	데이터베이스	20000
401	데이터베이스	20000
402	스포츠경영학	15000
502	자료구조	25000



학생번호	강좌이름
501	데이터베이스
401	데이터베이스
402	스포츠경영학
502	자료구조

강좌이름	수강료
데이터베이스	20000
스포츠경영학	15000
자료구조	25000

데이터 베이스 정규화

- BCNF 정규화 (BCNF)

= 정의

테이블에 대해 모든 결정자가 후보키가 되도록 테이블을 분해

과목ID	과목명	교수ID	교수 연락처
101	수학	1	010-1234-5678
102	영어	2	010-8765-4321



과목ID	과목명	교수ID
101	수학	1
102	영어	2

교수ID	교수 연락처
1	010-1234-5678
2	010-8765-4321

데이터 베이스 정규화

- INIT TABLE

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명	카드_ID
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동	c101
우리	102-999-101023	a12	실버	0.2	e1	홍길동	c102
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동	c103
국민	010-4567-4567	a21	로얄	1	e2	둘리	c201 c202

데이터 베이스 정규화

- INIT TABLE

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명	카드_ID
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동	c101
우리	102-999-101023	a12	실버	0.2	e1	홍길동	c102
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동	c103
국민	010-4567-4567	a21	로얄	1	e2	둘리	c201 c202

기준 키가 될 수 있는 컬럼 : 계좌_ID , { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명, 카드_ID

데이터 베이스 정규화

- 제 1 정규화 : 모든 컬럼의 값은 원자 값을 가진다.

데이터 베이스 정규화

- 제 1 정규화 : 모든 컬럼의 값은 원자 값을 가진다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명	카드_ID
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동	c101
우리	102-999-101023	a12	실버	0.2	e1	홍길동	c102
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동	c103
국민	010-4567-4567	a21	로얄	1	e2	둘리	c201
국민	010-4567-4567	a21	로얄	1	e2	둘리	c202

데이터 베이스 정규화

- 제 1 정규화 : 모든 컬럼의 값은 원자 값을 가진다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명	카드_ID
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동	c101
우리	102-999-101023	a12	실버	0.2	e1	홍길동	c102
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동	c103
국민	010-4567-4567	a21	로얄	1	e2	둘리	c201
국민	010-4567-4567	a21	로얄	1	e2	둘리	c202

기준 키가 될 수 있는 컬럼 : { 계좌_ID, 카드_ID }, { 은행명, 계좌번호, 카드_ID }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- 제 2 정규화 : 부분 함수 종속을 제거해 완전 함수 종속으로 만든다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명	카드_ID
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동	c101
우리	102-999-101023	a12	실버	0.2	e1	홍길동	c102
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동	c103
국민	010-4567-4567	a21	로얄	1	e2	둘리	c201
국민	010-4567-4567	a21	로얄	1	e2	둘리	c202

기준 키가 될 수 있는 컬럼 : { 계좌_ID, 카드_ID }, { 은행명, 계좌번호, 카드_ID }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- 제 2 정규화 : 부분 종속 함수를 제거해 완전 종속 함수로 만든다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동
우리	102-999-101023	a12	실버	0.2	e1	홍길동
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동
국민	010-4567-4567	a21	로얄	1	e2	둘리

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

계좌_ID	카드_ID
a11	c101
a12	c102
a13	c103
a21	c201
a21	c202

데이터 베이스 정규화

- 제 3 정규화 : 이행적 종속 관계를 없애라. ($X \rightarrow Y$ 이면, $Y \rightarrow Z$ 일 때, $X \rightarrow Z$ 이면 안된다.)

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동
우리	102-999-101023	a12	실버	0.2	e1	홍길동
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동
국민	010-4567-4567	a21	로얄	1	e2	둘리

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- 제 3 정규화 : 이행적 종속 관계를 없애라. ($X \rightarrow Y$ 이면, $Y \rightarrow Z$ 일 때, $X \rightarrow Z$ 이면 안된다.)

은행명	계좌번호	계좌_ID	등급	비율	사원_ID	사원명
우리	010-1234-1234	a11	브론즈	0.1	e1	홍길동
우리	102-999-101023	a12	실버	0.2	e1	홍길동
국민	010-1234-1234	a13	로얄	0.7	e1	홍길동
국민	010-4567-4567	a21	로얄	1	e2	둘리

계좌_ID \rightarrow 사원_ID

사원_ID \rightarrow 사원명

계좌_ID \rightarrow 사원명

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- 제 3 정규화 : 이행적 종속 관계를 없애라. ($X \rightarrow Y$ 이면, $Y \rightarrow Z$ 일 때, $X \rightarrow Z$ 이면 안된다.)

은행명	계좌번호	계좌_ID	등급	비율	사원_ID
우리	010-1234-1234	a11	브론즈	0.1	e1
우리	102-999-101023	a12	실버	0.2	e1
국민	010-1234-1234	a13	로얄	0.7	e1
국민	010-4567-4567	a21	로얄	1	e2

사원_ID	사원명
e1	홍길동
e2	둘리

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID

데이터 베이스 정규화

- 제 3 정규화 : 이행적 종속 관계를 없애라. ($X \rightarrow Y$ 이면, $Y \rightarrow Z$ 일 때, $X \rightarrow Z$ 이면 안된다.)

은행명	계좌번호	계좌_ID	등급	비율	사원_ID
우리	010-1234-1234	a11	브론즈	0.1	e1
우리	102-999-101023	a12	실버	0.2	e1
국민	010-1234-1234	a13	로얄	0.7	e1
국민	010-4567-4567	a21	로얄	1	e2

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

계좌_ID	카드_ID	사원_ID	사원명
a11	c101	e1	홍길동
a12	c102	e2	둘리
a13	c103		
a21	c201		
a21	c202		

데이터 베이스 정규화

- BCNF : 이 테이블의 모든 결정자가 후보키가 되도록 만든다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID
우리	010-1234-1234	a11	브론즈	0.1	e1
우리	102-999-101023	a12	실버	0.2	e1
국민	010-1234-1234	a13	로얄	0.7	e1
국민	010-4567-4567	a21	로얄	1	e2

국민은행 : 노말, 프레스티지, 로얄
우리은행 : 브론즈, 실버, 골드

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- BCNF : 이 테이블의 모든 결정자가 후보키가 되도록 만든다.

은행명	계좌번호	계좌_ID	등급	비율	사원_ID
우리	010-1234-1234	a11	브론즈	0.1	e1
우리	102-999-101023	a12	실버	0.2	e1
국민	010-1234-1234	a13	로얄	0.7	e1
국민	010-4567-4567	a21	로얄	1	e2

등급 → 은행

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

데이터 베이스 정규화

- BCNF : 이 테이블의 모든 결정자가 후보키가 되도록 만든다.

계좌번호	계좌_ID	등급	비율	사원_ID
010-1234-1234	a11	브론즈	0.1	e1
102-999-101023	a12	실버	0.2	e1
010-1234-1234	a13	로얄	0.7	e1
010-4567-4567	a21	로얄	1	e2

기준 키가 될 수 있는 컬럼 : { 계좌_ID }, { 은행명, 계좌번호 }

기준 키가 될 수 없는 컬럼 : 등급, 비율, 사원_ID, 사원명

등급 → 은행

등급	은행명
브론즈	우리
실버	우리
골드	우리
노말	국민
프레스티지	국민
로얄	국민

데이터 베이스 정규화

- 정규화 이후의 테이블

계좌번호	계좌_ID	등급	비율	사원_ID
010-1234-1234	a11	브론즈	0.1	e1
102-999-101023	a12	실버	0.2	e1
010-1234-1234	a13	로얄	0.7	e1
010-4567-4567	a21	로얄	1	e2

등급	은행명
브론즈	우리
실버	우리
골드	우리
노말	국민
프레스티지	국민
로얄	국민

사원_ID	사원명
e1	홍길동
e2	둘리

계좌_ID	카드_ID
a11	c101
a12	c102
a13	c103
a21	c201
a21	c202