

Chapter 5

기본키와 외래키

오 세 종

핵심 내용

- 기 본 키 (primary key)
- 외래 키 (foreign key)
- 조인 연산 (join operation)

1. 소개

- 무결성 규칙(Integrity rules)
 - DBMS 에게 실세계의 제약조건(ex. 무게>0) 을 알리기 위함
 - 데이터 값의 결정 : "현실의 반영(reflect reality)"
 - 데이터베이스의 정의는 무결성 규칙의 포함을 필요로 한다.
- Example : 부서-사원 데이터베이스를 위한 규칙 목록
 - 부서번호의 형식 : nn (nn 은 2자리 10진수)
 - 부서의 위치는 모두 미국내에 있어야 한다
 - 입사일자 : 1980-01-01 ~ 현재일자 이내여야 한다
 - 사장은 mgr 정보가 null 이어야 한다

무결성 규칙은 테이블을 생성할 때 지정해 준다
어떤 무결성 규칙은 구현할 수 없는 것도 있다

1. 소개

- 어떤 주어진 무결성 규칙은 :
 - 특정 데이터베이스에 대해서만 적용 된다 (database specific)
- 데이터베이스의 제한성 규칙에 추가하여, 두개의 일반적인 무결성 규칙을 갖는다.
 - 이 규칙은 특정 데이터베이스에만 적용되는 것이 아니라 모든 데이터베이스에 일반적으로 적용
 - (1) 후보키 (candidate key)
 - (2) 외래키 (foreign key)

2. 후보키

- 가장 기본적인 무결성 규칙
 - “테이블에 튜플이 입력될 때 중복이 없어야 한다”

customer

고객번호	고객명	전화번호	주민등록번호	주소지	등급
C001	홍길동	010-123-2324	801212-1214316	서울	A
C002	김철수	010-424-5343	810824-1134566	인천	A
C003	한나영	010-454-7364	891213-2321455	서울	B
C004	최민수	010-231-6143	940501-1565465	대구	C
C005	홍춘길	010-422-3452	930701-1836255	광주	B
C006	김나리	011-721-7771	950801-2156557	인천	C

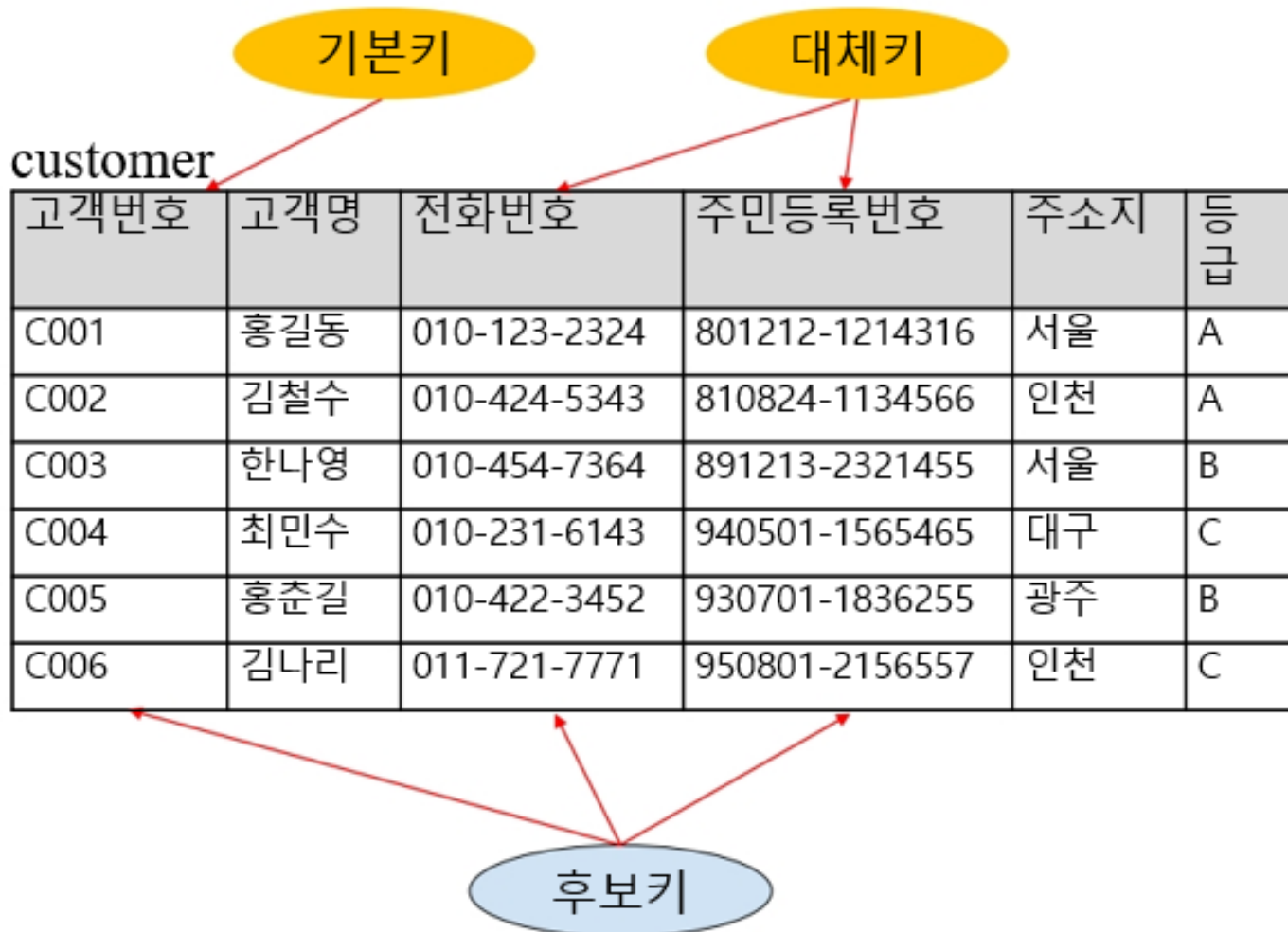


새로운 튜플 입력시 중복성 검사를 해야 한다. **HOW?**

C009	박성식	010-341-2411	900801-1155601	서울	C
------	-----	--------------	----------------	----	---

2. 후보키

- 모든 컬럼의 값을 비교해 볼수 는 없으니 기준이 되는 몇 개 또는 하나의 컬럼만 보고 중복성 여부를 판단하자 → 후보키(candidate key)



2. 후보키

- 후보키 (candidate key)
 - ◉ 중복성 여부를 판단할 수 있는 컬럼, 혹은 컬럼들의 집합
- 기본키(primary key)
 - ◉ 테이블에 대한 유일 식별자(unique identifier)
 - ◉ 후보키(candidate key) 개념의 특별한 경우
- 대체키(Alternate key)
 - ◉ 후보키 중 기본키로 지정되지 않은 다른 키

2. 후보키

- 테이블은 적어도 하나의 후보키를 갖는다
 - 테이블은 중복된 튜플을 갖지 않으므로.
 - 모든 속성의 조합이 후보키가 되는 경우도 있다.
- Note
 - 한 테이블은 하나 이상의 후보키를 가질 수 있다.
 - 단순(simple) 후보키 vs 복합(composite) 후보키

사원정보

A	사원번호	이름	주민등록번호
	S0001	신성호	701212-1063423
	S0002	이민식	720818-1023531
	S0003	김철수	740921-1125342

사원별 거래처

B	사원번호	거래처	거래금액
	S0001	대성물산	5,000
	S0001	신수전자	6,000
	S0002	신수전자	4,000

2. 후보키

- 정리
 - 기본키는 튜플의 중복성을 판별하는데 기준이 되는 컬럼 또는 컬럼들의 집합
- 후보키 중 어느 것이 기본키로 선택되어 지는가?
 - 정해진 기준은 없다
 - 단순키와 복합키가 있다면 단순키를 기본키로 선택

customer

고객번호	고객명	전화번호	주민등록번호	주소지	등급
C001	홍길동	010-123-2324	801212-1214316	서울	A
C002	김철수	010-424-5343	810824-1134566	인천	A
C003	한나영	010-454-7364	891213-2321455	서울	B
C004	최민수	010-231-6143	940501-1565465	대구	C
C005	홍준길	010-422-3452	930701-1836255	광주	B
C006	김나리	011-721-7771	950801-2156557	인천	C

3. 외래키

- 외래키(foreign key)의 용어정의
 - R2라는 기본 테이블 에서 R2의 속성 집합의 부분집합중에서 다음을
 - 만족 시키는 FK 를 R2의 외래키(foreign key) 라고 한다.
- CK를 후보키로 갖는 기본 테이블 R1이 이미 존재하고 있어야 한다. (R1과 R2가 서로 다를 필요는 없다)
- 주어진 어느 시간이라도, R2의 현재 값의 각 FK의 값은 R1의 현재 값의 어떠한 튜플인 CK 의 값과 같아야 한다

3. 외래키

- 외래키의 예

DEPT	기본키		
	DEPTNO	DNAME	BUDGET
	D1	마케팅부	10M
	D2	개발부	12M
	D3	연구부	5M

EMP	기본키		외래키	
	EMPNO	ENAME	DEPTNO	SALARY
	E1	이몽룡	D1	40K
	E2	장길산	D1	42K
	E3	허참봉	D2	30K
	E4	성춘향	D2	35K

참조되는
테이블
(부모)

참조하는
테이블
(자식)

3. 외래키

- **요점**

- 외래키는 후보키처럼 속성의 집합으로 정의 된다.
- 정의에 의해, 주어진 외래키의 모든 값은 참조하는 후보키와 부합되는 값을 가져야 한다. (그 역은 성립되지 않음)
- 주어진 외래키가 복합키이면 참조하는 후보키도 복합키임을 의미
- 기본키의 일부가 외래 키인 경우도 있다.
- 자식테이블의 외래키는 부모 테이블의 기본키를 참조하도록 되어 있다

DEPT	DEPT#	DNAME	BUDGET
	<i>primary key</i>		

부모 테이블

EMP	EMP#	ENAME	DEPT#	SALARY
	<i>primary key</i>		<i>foreign key</i>	


자식 테이블

3. 외래키

- 자기 참조(*Self-Referencing*)
 - 외래키 정의 관계에 있는 테이블 R1 과 R2 가 동일한 테이블 일 때

EMP

기본키	EMPNO	ENAME	DEPT#	외래키	MGR	SALARY
	E1	이몽룡	D1		E2	40K
	E2	장길산	D1			42K
	E3	허참봉	D2		E1	30K
	E4	성춘향	D2		E3	35K



3. 외래키

- 참조 무결성 규칙(referential integrity rule)
 - 데이터베이스는 부합되지 않는 어떠한 외래키 값도 포함해서는 안된다.
 - 만일 B 가 A를 참조 한다면, A 는 반드시 존재해야 한다
 - 외래키를 지원하는 것과 참조 무결성을 지원하는 것은 같은 의미

4. 외래키 규칙


- 참조 무결성 규칙 (referential integrity rule)
 - 데이터베이스가 유지해야 할 정상적 상태
- 참조 무결성의 수행
 - 옳지 못한 연산이 수행될 경우 시스템이 거부
 - 추가적인 보상 연산을 통하여 시스템의 무결성 유지.(ex.cascading delete)
- 데이터베이스 설계자(database designer) 는
 - 어떤 연산이 거부되어야 하는지를 정의
 - 어떤 보상연산이 수행되어야 하는지를 정의

4. 외래키 규칙

- 외래키 참조의 대상(target)이 되는 후보키를 갱신하려고 할 때 어떻게 할 것 인가?
 - 제한(RESTRICTED) : 부합(match)되는 후보키가 없을 경우에만 삭제가
능
 - 연쇄(CASCADES) : 부합 (match) 되는 모든 후보키도 함께 갱신
 - Nullify
 - Nothing

DEPT	DEPT#	DNAME	BUDGET
	D1	마케팅부	10M
	D2	개발부	12M
	D3	연구부	5M

EMP	EMP#	ENAME	DEPT#	SALARY
	E1	이몽룡	D1	40K
	E2	장길산	D1	42K
	E3	허참봉	D2	30K
	E4	성춘향	D2	35K



5. Null

- 널(null)
 - ◉ 부재정보(*missing information*)를 다루는 기본적인 방법
 - ◉ "생일이 알려져 있지 않음", "앞으로 알려질 것임"
 - ◉ 자료형이 없다
- 널은 공백(blank) 또는 제로(zero)와는 다르다

6. 후보키와 Null

- 개체 무결성 규칙(entity integrity rule)
 - 기본 테이블의 기본키의 어떤 성분도 널(null)일 수 없다
- 근거(rationale)
 - 기본 테이블은 현실세계의 개체들에 해당 한다
 - 현실세계의 개체들은 식별가능 하다(*distinguishable & identifiable*)
 - 기본키는 관계형 모델에서 유일 식별성(unique identification)을 수행 한다
 - 식별되지 않는 개체는 존재하지 않는다

** 동일한 이유로 복합 기본키의 *부분적인* 널 값도 방지 되어야 한다

7. 조인 연산

- 조인(join) 연산이란
 - 두개 또는 두개 이상의 테이블을 연결하여 하나의 테이블처럼 보이게 하는 연산
- 예제 테이블

emp

empid	ename	deptno
E1	홍길동	D1
E2	김철수	D1
E3	한나영	D2

dept


deptno	dname
D1	영업부
D2	관리부
D3	생산부

7. 조인 연산

SELECT * FROM emp, dept

emp

dept

empid	ename	deptno		deptno	dname
E1	홍길동	D1		D1	영업부
E2	김철수	D1		D2	관리부
E3	한나영	D2		D3	생산부

empid	ename	deptno	deptno	dname
E1	홍길동	D1	D1	영업부
E1	홍길동	D1	D2	관리부
E1	홍길동	D1	D3	생산부
E2	김철수	D1	D1	영업부
E2	김철수	D1	D2	관리부
E2	김철수	D1	D3	생산부
E3	한나영	D2	D1	영업부
E3	한나영	D2	D2	관리부
E3	한나영	D2	D3	생산부

7. 조인 연산

```
SELECT *  
FROM emp, dept  
WHERE emp.deptno = dept.deptno
```

emp

dept

empid	ename	deptno		deptno	dname
E1	홍길동	D1		D1	영업부
E2	김철수	D1		D2	관리부
E3	한나영	D2		D3	생산부

empid	ename	deptno	deptno	dname
E1	홍길동	D1	D1	영업부
E2	김철수	D1	D1	영업부
E3	한나영	D2	D2	관리부

7. 조인 연산

```
SELECT emp.empid, emp.ename, emp.deptno, dept.dname
FROM emp, dept
WHERE emp.deptno = dept.deptno
```

```
SELECT empid, ename, emp.deptno, dname
FROM emp, dept
WHERE emp.deptno = dept.deptno
```

```
SELECT empid, ename, e.deptno, dname
FROM emp e, dept d
WHERE e.deptno = d.deptno
```

empid	ename	deptno	dname
E1	홍길동	D1	영업부
E2	김철수	D1	영업부
E3	한나영	D2	관리부

[연습]

- 모든 사원의 사원번호, 사원이름, 부서명, 급여액을 보이시오