



1. 데이터베이스 기본기

데이터베이스 용어 (1)

| A | B | C | D | E | F |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |



: 애트리뷰트

-차수(Degree) : 애트리뷰트의 갯수, -스키마 : 애트리뷰트들의 집합!



: 튜플

-카디널리티 : 튜플의 갯수, -인스턴스 : 튜플들의 집합!



: 도메인 -> 하나의 열에 있는 값들의 집합!

저 위에 있는 표 전체 -> 릴레이션, 테이블

- 1) 튜플의 각 애트리뷰트는 원자값만 가져야함!!! 내용물이 하나이상이면 안된다.
- 2) 릴레이션이 튜플들의 집합이기 때문에 동일한 튜플이 두개 이상 존재하지 않는다.
- 3) 당연하게도 도메인 안에선 전부다 같은 타입의 값을 가져야겠죠?

데이터베이스 용어 (2)

| 주민번호 | 이름 | 나이 |
|----------------|-----|----|
| 971111-1234567 | 길동이 | 22 |
| 961111-8888888 | 길동이 | 23 |
| 971111-7777777 | 이길동 | 22 |

1. 저기에서 기본키는?

2. 후보키? 대체키? 슈퍼키? 기본키??

- 슈퍼키는 조금이라도 튜플들을 구분할수있는 요소가 있는 애트리뷰트,애트리뷰트의 집합
- !부분릴레이션!에서 튜플을 구별가능한 키
- 후보키는 !릴레이션!에서 각 튜플을 고유하게 식별할 수 있는 애트리뷰트의 모임
- 기본키는 후보키중에서 하나의 애트리뷰트를 택해서 정한것 이다.

기본키의 조건 1)항상 고유한 값을 가지는가?

2)항상 널값을 갖지 않는가?

3)값이 변경될 가능성이 높은 애트리뷰트인가? (가능성이 낮아야 한다)

4) 가능하면 복합 기본키로 선택하지 않는다.(기본키가 하나이상이지 말자.)

데이터베이스 용어 (2)

- 대체 키(Alternate Key)

:기본키가 될뻔한 키

후보키에서 기본키가 아닌 다른키라고 생각하면 됩니다.

- 키들의 구도

수퍼 키 > 후보키 (기본키 + 대체키)

수퍼키는 전부다 포괄하니까 슈퍼라고 생각하면 편합니다.

- 외래 키(Foreign Key)

아무 릴레이션의 기본키를 참조하는 키

- 1) 다른 릴레이션의 기본키를 참조하는 키
- 2) 자체 릴레이션의 기본 키를 참조하는 키
- 3) 기본키의 구성요소가 될수 있다.

무결성 제약조건?

- 데이터 무결성은 데이터의 정확성 또는 유효성을 의미
- 무결성 제약조건의 목적은 일관된 데이터베이스 상태를 정의하는 규칙을 묵시적으로 또는 명시적으로 정의하는 것.
- 쉽게 말해서 어기면 안되는 규칙입니다.
프로그래밍에서도 Integer형에는 정수만 넣는거처럼 여기서도 데이터들의 규칙이 있습니다.

무결성 제약조건

- ◉ 도메인 제약조건

: 한 애트리뷰트의 값들이 같은 데이터 형식을 가지고 있어야한다.

- ◉ 키 제약조건

: 기본키로 정의 하거나 UNIQUE를 명시한 애트리뷰트끼리는 중복이 없어야 한다.

- ◉ 기본키와 엔티티 무결성 제약조건

: 릴레이션의 기본키를 구성하는 어떤 애트리뷰트도 널값을 갖지 못한다.

※ 대체키에는 해당이 되지 않는다!

- ◉ 외래키와 참조 무결성 제약조건

1) 외래 키의 값은 한 릴레이션의 어떤 튜플의 기본 키 값과 같다.

2) 외래 키가 자신을 포함하고 있는 릴레이션의 기본 키를 구성하지 않으면 널 값을 가질 수 있다.

※ 엔티티란?

DB를 구성할때의 추상적인 개체다위를 뜻하는거

무결성 제약조건의 유지

EMPLOYEE

| EMPNO | EMPNAME | DNO |
|-------|---------|-----|
| 2106 | 김창섭 | 2 |
| 3426 | 박영권 | 3 |
| 3011 | 이수민 | 1 |

DNO는
DEPTNO를
참조

DEPARTMENT

| DEPTNO | DEPTNAME | FLOOR |
|--------|----------|-------|
| 1 | 안내 | 1 |
| 2 | R&D | 8 |
| 3 | 마케팅 | 7 |
| 4 | 운영 | 10 |

- ★ 여기서의 삽입 삭제 수정은 만약 했을때 무결성을 위반할 가능성을 설명하는것
- 삽입 : 기존에 있는 테이블의 무결성 조건을 깨지 않으면 된다는것.(도메인, 키, 엔티티)
- 삭제 : 이넘은 도메인, 키, 엔티티 무결성을 깨진 않는데 만약 삭제했을때 EMPLOYEE에서 참조키가 가리키는 값이 사라질수있음
 - 1)제한 : EMPLOYEE에서 참조하는 튜플이 사라진다면 강 거절해버림
 - 2)연쇄 : EMPLOYEE에서 참조하는 튜플이 사라진다면 참조하고 있는 외래키를 같이 삭제해버림
 - 3)널값, 디폴트값 : EMPLOYEE에서 참조하는 튜플이 사라진다면참조하고 있는 외래키를 널이나 디폴트값으로 바꿈
- 수정 : 수정은 하나의 튜플을 삭제하고 새로운 튜플을 삽입하는것과 유사하다. 그래서 삽입과 삭제의 무결성 위반 가능성이 여기서도 똑같이 나타날수있다. (왜냐면 삭제와 삽입을 동시에 하니까!)

데이터 명령어

- 데이터 조작어 (Data Manipulation Language)
: 튜플을 입력(INSERT), 수정(UPDATE), 삭제(DELETE)할때 쓰는 명령어
- 데이터 정의어 (Data Definition Language)
: DB의 스키마 객체를 생성(CREATE), 변경(ALTER), 삭제(DROP),
이름변경(RENAME), DB저장공간삭제(TRUNCATE) 할때 쓰는 명령어
- 데이터 제어어 (Data Control Language)
: DB의 대한 권한 부여(GRANT), 취소(REVOKE) 할때 쓰는 명령어
- 트랜잭션 처리어(Transaction Control Language)
: 트랜잭션의 정상적인 종료(COMMIT), 취소(ROLLBACK), 임시저장점 지정
(SAVEPOINT)

제약조건을 한번 걸어보자!!

- SQL문에서 제약조건 걸기

CREATE TABLE EMPLOYEE

```
(  EMPNO      NUMBER      NOT NULL,  
   EMPNAME    CHAR(10)    UNIQUE,  
   TITLE      CHAR(10)    DEFAULT '사원',  
   MANAGER     NUMBER,  
   SALARY      NUMBER     CHECK (SALARY < 6000000),  
   DNO         NUMBER     CHECK (DNO IN (1,2,3,4,5,6)) DEFAULT 1,  
   PRIMARY KEY (EMPNO),  
   FOREIGN KEY (MANAGER) REFERENCES EMPLOYEE (EMPNO),  
   FOREIGN KEY (DNO)    REFERENCES DEPARTMENT (DEPTNO)  
                                ON DELETE CASCADE );
```

굵은 줄 전부 제약조건 명시한것.

관계대수 조아

- 관계대수란?

: 어떻게 질의를 수행할 것인가 명시하는 절차적 언어
결과적으로 보면 주어진 릴레이션에서 새로운 릴레이션을 생성한다.

- 관계대수들을 소개합니다.

선택(selection) σ , 프로젝션(Projection) π ,
조인(Join) \bowtie (애도..), 카티션의 곱 \times , 디비전(Division) \div