



맹준영_1장

목표

- 관계형 모델의 개요
- SQL이 무엇인가
- 관계형 모델이 어떤 것인지 틀 잡기
- SQL이 관계형 모델에 어떻게 대입되는지, 어떻게 다른지

▼ 1.2 관계형 모델

▼ 관계형 모델



데이터 모델 (Data Model) = 데이터의 표현 방식
ex) Key-Value 이용한 데이터 표현 모델, 관계형 모델

- 관계형 모델 : 실제 세계 데이터 → “관계(Relation)” 개념 표현 모델
- 릴레이션 = SQL 테이블, 제목(데이터베이스 스키마, 컬럼 집합) + 본체
 - 제목 : 속성이 0개 이상 모인 집합 (이름 + 데이터 타입)
 - 본체 : 속성값의 집합, 튜플의 집합, 제목과 연관된 속성값을 가짐

관계형 모델 VS SQL

관계형 모델	SQL
릴레이션	테이블
튜플	행
속성	컬럼

▼ 집합



집합 = 요소의 모임

[조건]

1. 요소 자체로 요소의 집합 포함 여부 파악 가능해야함 → 집합의 성질에 맞는 요소만 집합에 포함되야 하는 것을 의미
2. 요소의 중복 X

- 릴레이션 = 튜플의 집합, 즉 튜플이 모여져 릴레이션을 이룸
- 집합의 조건 관점에서 본 릴레이션
 - 집합 성질에 맞는 요소만 집합에 포함 → 각 튜플이 릴레이션의 속성에 맞는 값으로 구성되어야 한다.
 - 요소의 중복 X → 릴레이션은 중복된 튜플을 갖지 않는다.

관계형 모델에서의 NULL

NULL = 특정한 값이 아닌 요소가 무엇을 표현하는지 모르는 불명확함을 표현하는 것
 → 집합 성질에 맞는 요소만 집합에 포함되야 한다는 조건을 위반(?)

NULL은 값이 불명확하기 때문에 특정 속성값을 이용한 릴레이션 연산에서는 연산 결과가 달라질 수 있기 때문에 관계형 모델에 적합하지 않다.

▼ 릴레이션의 연산



릴레이션의 연산 = Closure

릴레이션을 이용한 연산의 결과는 릴레이션이다.

따라서, 릴레이션 연산 결과를 다른 릴레이션과 계속해서 연산해 나가는 것이 가능하다.

Closure (클로저) : 연산의 입력과 출력이 같은 데이터 구조를 갖는 성질

- 제한(Restrict) : 특정 조건에 맞는 튜플을 포함한 릴레이션 부분집합 반환 (튜플 단위 연산, 릴레이션 전체, 공집합 가능) → SQL의 WHERE절
- 프로젝션(Projection) : 특정 속성만 포함하는 릴레이션 반환 (속성 단위 연산), 중복 튜플 발생 시 중복 제거 → SQL의 SELECT절
- 확장(Extend) : 프로젝션과 반대, 기존 속성을 통해 새로운 속성으로 확장
- 속성명 변경(Rename) : 속성 이름 변경, 확장(Extend) 연산과 함께 주로 사용 → 확장 속성의 이름 변경
- 합집합 (Union) : 두 릴레이션의 모든 튜플 합침 → 중복 제거
- 교집합 (Intersect) : 두 릴레이션에서 공통된 튜플 반환
- 차집합 (Difference) : 한쪽 릴레이션에서 다른쪽 릴레이션 튜플과 비교해 공통된 부분을 제외한 남은 튜플 반환
- 곱집합 (Product) : 두 개의 릴레이션 튜플을 각각 조합한 하나의 릴레이션 반환
 ex) A = 3개 튜플, B = 2개 튜플 - 곱집합 연산 = 6개 튜플 반환
 → SQL의 FROM절
- 결합 (Join) : 두 개의 릴레이션 공통된 속성값 튜플을 조합한 릴레이션 반환, 공통된 속성 갖지 않는 튜플 제외

▼ 데이터 형식



관계형 모델에서의 데이터 형식은 “어떻게 데이터를 사용(표현) 할지에 대한 고려”이다.

- 도메인 : 관계형 모델에서의 데이터 형식
 - 속성값에 사용되는 요소의 집합
 - 속성에 대한 요소 집합값만 속성에 사용 가능
 - ex) 성별 속성에 대한 도메인은 (남자, 여자) → 성별 속성은 이 두가지 요소를 통해 표현 가능

▼ 1.3 SQL에서 릴레이션 조작

- SELECT : Projection + Product + Restrict
= 테이블 행 조회

```
SELECT 조회 컬럼(속성) 목록 -> Projection
FROM 테이블(릴레이션) 목록 -> Product
WHERE 검색 조건 -> Restrict
```

- INSERT : (구)릴레이션에 새로운 튜플 추가 → (신)릴레이션을 다시 반환받는 연산
= 테이블에 행 추가
- DELETE : (구)릴레이션에 조건절에 맞는 튜플 삭제 → (신)릴레이션을 반환받는 연산
= 테이블의 행 삭제
- UPDATE : Delete And Insert 방식의 연산
(구)릴레이션 조건 절에 대한 튜플 삭제 → (신)릴레이션을 반환 → 다시 튜플을 추가 → (최신)릴레이션을 반환
= 테이블의 데이터 수정

▼ 1.4 SQL에는 있고 관계형 모델에는 없는 것

릴레이션	SQL
튜플 중복 X	제약 조건에 따른 중복 행 O
요소의 순서 X	컬럼에 따른 순서 O → Rownum, Order By 사용시 순서 기반 쿼리 지양
트랜잭션 없음	트랜잭션 있음
NULL 포함 불가	NULL 가능

질문?



P16. Join 연산의 특수한 예 → 교집합, 곱집합