

SQL (Structured Query Language)

SQL 기능

- 데이터 정의어 (DDL)
- 데이터 조작어 (DML) CRUD
- 뷰 정의어
- 보안과 권한 관리
- 무결성 제약 조건 명시
- 트랜잭션 제어

Embedded SQL (DSL : Data Server Language)은 Java, COBOL, C/C++ (Host language)에 Embedding 가능

CREATE - 테이블 생성

DROP - 테이블 삭제

INSERT - 행 삽입

DELETE - 행 삭제

DDL, DML

트랜잭션 지원하는 엔진 - InnoDB

애트리뷰트 데이터 타입

숫자

- 정수형 : INTEGER, INT, SMALLINT
- 부동소수점수 : FLOAT, REAL, DOUBLE PRECISION
- 형식화된 수 : DECIMAL(i,j), NUMERIC(i,j) (i : 숫자의 총 개수. 유효숫자, j : 소수점 뒤에 오는 숫자의 개수). DECIMAL(5,2) : -999.99 ~ 999.99

문자열

- 고정 길이 - CHAR(n)
- 가변 길이 - VARCHAR(n)
- CLOB : Character Large Object

비트열

- 고정길이 - BIT(n)
- 가변길이 - BIT VARYING(n)
- BLOB - Binary Large Object

BOOLEAN (TRUE/FALSE)

DATE, TIME, TIMESTAMP 데이터 타입을 추가로 가짐

- DATE :
 - yyyy-mm-dd 형식으로 year-month-day을 표현
 - DATE '2008-09-27'
- TIME
 - hh:mm:ss 형식으로 hour:minute:second을 표현함
 - TIME '09:12:27'
- TIMESTAMP
 - DATE와 TIME 구성요소를 포함
 - TIMESTAMP '2002-09-27 09:12:47 648302'

SQL에서 기본제약조건의 명시

- 애틀리뷰트 제약조건과 디폴트값 명시
 - 애틀리뷰트에 NULL 값의 허용여부 명시 : NOT NULL
 - 애틀리뷰트에 디폴트 값 지정 : DEFAULT <값>
 - 애틀리뷰트나 도메인 값을 특정한 값(들)으로 한정 : CHECK (값의 범위)

- 키와 참조 무결성 제약조건의 명시

- CREATE TABLE 명령에서 PRIMARY KEY 절은 릴레이션의 기본 키를 구성하는 하나 이상의 애트리뷰트들을 명시하고, UNIQUE 절은 대체키(또는 보조키)를 명시하며, FOREIGN KEY 절에서는 참조 무결성을 지정함
- 외래 키(FOREIGN KEY)를 정의할 때, 참조 무결성의 위반시 취할 동작을 명시할 수 있음. 동작의 종류에는 SET NULL, CASCADE, SET DEFAULT가 있으며, 위반의 종류를 나타내는 ON DELETE나 ON UPDATE와 함께 사용
- SQL 시스템들은 참조 무결성과 기본 키를 지정
- 초기 SQL에서는 기본 키를 지정하는 기능이 없고 CREATE UNIQUE INDEX 명령을 통해 지정하고 있음

SQL에서 기본 검색 질의

- SQL은 데이터베이스로부터 정보를 검색하는 문장을 가짐
- 관계 대수의 SELECT 연산과는 무관
- SQL과 관계모델의 중요한 차이점
- SQL의 테이블(릴레이션)은 모든 애트리뷰트 값이 동일한 튜플을 하나 이상 가질 수 있음
- 따라서, SQL 릴레이션(테이블)은 튜플의 집합이 아니라 튜플의 다중집합(multi-set or bag)임
- 키 제약조건을 선언하거나 DISTINCT 선택사항을 사용하여 SQL 릴레이션들을 집합으로 제한

간단한 SQL 질의들

'Research'부서에서 일하는 모든 종업원들의 이름과 주소를 검색하시오

SELECT FNAME, LNAME, ADDRESS

FROM EMPLOYEE, DEPARTMENT EMPLOYEE X DEPARTMENT (카티션 곱)

WHERE DNAME='Research' AND DNUMBER=DNO

SQL 작성시 릴레이션 이름 다음에 (.) 을 두고 애트리뷰트 이름을 명시함

Example) EMPLOYEE.LNAME, DEPARTMENT.DNAME

별명(ALIAS)

WHERE 절의 생략

*의 사용

DISTINCT 사용

집합 연산

- 합집합(UNION)연산, 차집합(EXCEPT) 연산, 교집합(INTERSECT) 연산 제공
- UNION ALL, EXCEPT ALL, INTERSECT ALL 연산은 다중집합 연산임.
- Example) $A=\{1,1,2,2,2,3\}$ $B=\{1,2,2,3,3,4\}$
- $A-B \text{ except all} = \{1,2\}$
- $A \text{ intersect all } B = \{1,2,2,3\}$

부분 문자열 비교

% : wild string

_ : wild character

%sql select ' ' like '%'; True 1

%sql select 'a' like '_'; True 1

%sql select 'ab' like '_'; False 0