

8장. 데이터베이스 언어 SQL

SQL 소개

- SQL (Structured Query Language)
 - 관계 데이터베이스를 위한 표준 질의어
 - 1974년 IBM 연구소에서 시스템 R을 위해 만들어진 언어

SQL의 분류

1. 데이터 정의어 (DDL; Data Definition Language)
 - 테이블을 생성/변경/제거
2. 데이터 조작어 (DML; Data Manipulation Language)
 - 테이블에 데이터를 삽입/수정/삭제/검색
3. 데이터 제어어 (DCL; Data Control Language)
 - 보안을 위해 사용자별 데이터 접근/사용 권한 관리

데이터 정의/조작어를 중심으로 강의 진행

유형	명령문
DDL - 객체 생성 및 변경 시 사용	CREATE (테이블 생성) ALTER (테이블 변경) DROP (테이블 삭제)
DML - 데이터 변경 시 사용	SELECT (데이터 검색) INSERT (데이터 입력) UPDATE (데이터 수정) DELETE (데이터 삭제)

예시

❖ 질의에 사용할 판매 데이터베이스 : 고객 릴레이션

고객 릴레이션

고객아이디	고객이름	나이	등급	직업	적립금
apple	정소화	20	gold	학생	1000
banana	김선우	25	vip	간호사	2500
carrot	고명석	28	gold	교사	4500
orange	김용욱	22	silver	학생	0
melon	성원용	35	gold	회사원	5000
peach	오형준	NULL	silver	의사	300
pear	채광주	31	silver	회사원	500

❖ 질의에 사용할 판매 데이터베이스 : 제품 릴레이션

제품 릴레이션

제품번호	제품명	재고량	단가	제조업체
p01	그냥만두	5000	4500	대한식품
p02	매운짬면	2500	5500	민국푸드
p03	쿵떡파이	3600	2600	한빛제과
p04	맛난초콜렛	1250	2500	한빛제과
p05	얼큰라면	2200	1200	대한식품
p06	통통우동	1000	1550	민국푸드
p07	달콤비스킷	1650	1500	한빛제과

❖ 질의에 사용할 판매 데이터베이스 : 주문 릴레이션

주문 릴레이션

주문번호	주문고객	주문제품	수량	배송지	주문일자
o01	apple	p03	10	서울시 마포구	2013-01-01
o02	melon	p01	5	인천시 계양구	2013-01-10
o03	banana	p06	45	경기도 부천시	2013-01-11
o04	carrot	p02	8	부산시 금정구	2013-02-01
o05	melon	p06	36	경기도 용인시	2013-02-20
o06	banana	p01	19	충청북도 보은군	2013-03-02
o07	apple	p03	22	서울시 영등포구	2013-03-15
o08	pear	p02	50	강원도 춘천시	2013-04-10
o09	banana	p04	15	전라남도 목포시	2013-04-11
o10	carrot	p03	20	경기도 안양시	2013-05-22

DDL

- 데이터 정의 기능
 - 테이블 생성 / 변경 / 제거
 - 테이블 생성 : CREATE TABLE
 - 테이블 변경 : ALTER TABLE
 - 테이블 제거 : DROP TABLE

CREATE TABLE

```
CREATE TABLE 테이블_이름 (  
    속성_이름 데이터_타입 [NOT_NULL] [DEFAULT] -- 테이블을 구성하는 각 속성 (이름, 타입,  
    제약사항)  
    [PRIMARY KEY (속성_리스트)] -- 기본키  
    [UNIQUE (속성_리스트)] -- 대체키 (Access에서는 동작 X)  
    [FOREIGN KEY (속성_리스트) REFERENCES 테이블_이름(속성_리스트)] [ON DELETE 옵션]  
    [ON UPDATE 옵션] -- 외래키  
    [CONSTRAINT 이름] [CHECK (조건)] -- 데이터 무결성을 위한 제약조건 (Access에서는 동  
    작 X)  
);
```

[] 안의 내용은 생략 가능

;;로 문장의 끝 표시 / 대소문자 구분 X

- 속성 이름 / PRIMARY KEY / FOREIGN KEY에 집중하여 사용
- 여기서 제약사항은 NULL, DEFAULT 같은 것들을 이야기 함

속성 정의

2번째 줄

- 테이블을 구성하는 각 속성의 데이터 타입 선택 -> NULL 값 허용 여부, 기본값 필요 여부 결정
- NOT NULL : NULL을 허용 X
- DEFAULT : 속성의 기본 값을 지정

예시)

```
고객아이디 VARCHAR(20) NOT NULL
적립금 INT DEFAULT 0
담당자 VARCHAR(10) DEFAULT '방경아'
```

키 정의

3, 4, 5번째 줄

- 기본키 - NULL 값이 허용 X
- 대체키 - 유일성을 가지며 NULL 값이 허용
- 외래키 - 외래키가 어떤 테이블의 무슨 속성을 참조하는지 REFERENCES와 함께 제시

예시)

```
PRIMARY KEY(고객아이디)
PRIMARY KEY(주문고객, 주문제품)
UNIQUE(고객이름)
FOREIGN KEY(소속부서) REFERENCES 부서(부서번호) -- 소속부서는 부서 테이블의 부서번호를 참조한다는 뜻
```

데이터 무결성 제약조건 정의

6번째 줄

- CHECK
 - 테이블에 정확하고 유효한 데이터를 유지하기 위해, 특정 속성에 대한 제약조건을 지정
 - CONSTRAINT 키워드와 함께 고유 이름 부여 가능

예시)

```
CHECK (재고량 >= 0 AND 재고량 <= 10000)
CONSTRAINT CHK_CPY CHECK (제조업체 = '한빛제과')
```

예제

- 고객 테이블 생성

고객 테이블은 고객아이디, 고객이름, 나이, 등급, 직업, 적립금 속성으로 구성 (고객아이디 속성이 기본키)

고객이름과 등급 속성은 반드시 입력

적립금 속성은 입력하지 않으면 0이 기본으로 입력

```
CREATE TABLE 고객 (  
    고객아이디 VARCHAR(20) NOT NULL, -- 기본키 (NOT NULL)  
    고객이름 VARCHAR(10) NOT NULL,  
    나이 INT,  
    등급 VARCHAR(10) NOT NULL,  
    직업 VARCHAR(10),  
    적립금 INT DEFAULT 0,  
    PRIMARY KEY(고객아이디)  
);
```

- coma 필요

- 제품 테이블 생성

제품 테이블은 제품번호, 제품명, 재고량, 단가, 제조업체 속성으로 구성 (제품번호 속성이 기본키)

재고량이 항상 0개 이상, 10,000개 이하 유지

```
CREATE TABLE 제품 (  
    제품번호 CHAR(3) NOT NULL,  
    제품명 VARCHAR(20),  
    재고량 INT,  
    단가 INT,  
    제조업체 VARCHAR(20),  
    PRIMARY KEY(제품번호),  
    CHECK (재고량 >= 0 AND 재고량 <=10000)  
);
```

- CHAR는 고정 길이 공간 / VARCHAR는 가변 길이 공간

- 주문 테이블 생성

주문 테이블은 주문번호, 주문고객, 주문제품, 수량, 배송지, 주문일자 속성으로 구성 (주문번호 속성이 기본키)

주문고객 속성이 고객 테이블의 고객아이디 속성을 참조하는 외래키

주문제품 속성이 제품 테이블의 제품번호 속성을 참조하는 외래키

```
CREATE TABLE 주문 (  
    주문번호 CHAR(3) NOT NULL,  
    주문고객 VARCHAR(20),  
    주문제품 CHAR(3),  
    수량 INT,  
    배송지 VARCHAR(30),  
    주문일자 DATETIME,  
    PRIMARY KEY(주문번호),  
    FOREIGN KEY(주문고객) REFERENCES 고객(고객아이디),  
    FOREIGN KEY(주문제품) REFERENCES 제품(제품번호)  
);
```

ALTER TABLE

새로운 속성 추가

```
ALTER TABLE 테이블_이름 ADD 속성_이름 데이터_타입 [NOT NULL] [DEFAULT 기본값];
```

기존 속성 삭제

```
ALTER TABLE 테이블_이름 DROP 속성_이름 CASCADE | RESTRICT;
```

- CASCADE - 삭제할 속성과 관련된 제약조건이나 참조하는 다른 속성을 함께 삭제
- RESTRICT - 삭제할 속성과 관련된 제약조건이나 참조하는 다른 속성이 존재하면 삭제 거부

예제

- 속성 추가

고객 테이블에 가입날짜 속성 추가

```
ALTER TABLE 고객 ADD 가입날짜 DATETIME;
```

- 속성 삭제

고객 테이블의 등급 속성을 삭제 - 관련된 제약조건이나 등급 속성을 참조하는 다른 속성도 함께 삭제

```
ALTER TABLE 고객 DROP 등급 CASCADE;
```

RENAME

추가적으로 찾아봄

```
ALTER TABLE old_table RENAME new_table; -- 단일 테이블 이름 변경  
RENAME TABLE old_table TO new_table; -- 단일/다수 테이블 이름 변경
```

DROP TABLE

```
DRIP TABLE 테이블_이름 CASCADE | RESTRICT;
```

- CASCADE - 제거할 테이블을 참조하는 다른 테이블도 함께 제거
- RESTRICT - 제거할 테이블을 참조하는 다른 테이블이 존재하면 제거 거부

예제

고객 테이블을 삭제, 고객 테이블을 참조하는 다른 테이블이 존재하면 삭제가 되지 않음

```
DRIP TABLE 고객 RESTRICT;
```