



# SQL

✓ 원리를 알면 IT가 맛있다

## SQL for Beginners



chapter 01.

---

# 데이터베이스 개요

- 데이터베이스 등장배경
- 데이터베이스 관리 시스템 ( DBMS )
- 데이터베이스 정의
- 관계형 데이터 모델 (관계형 데이터베이스 )
- SQL 개요
- SQL 종류

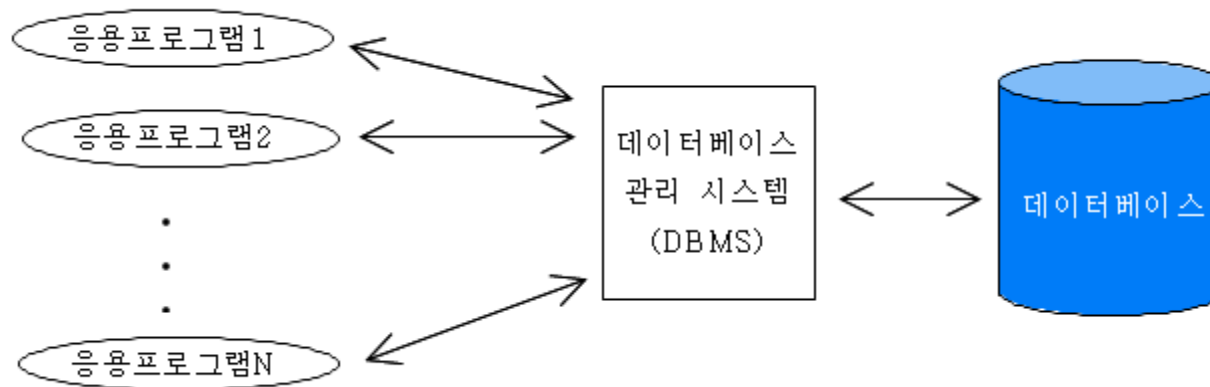
- 우리가 살고 있는 세상은 ‘정보화 사회’ 또는 ‘지식기반사회’이다. 오랜 세월 동안 축적한 대량의 데이터로부터 의미 있는 지식 정보를 효과적으로 추출하거나 새로운 지식을 재생산하는데 활용하기 위한 환경을 의미한다.
- 이러한 지식기반사회에서 가장 핵심적인 요소는 컴퓨터에 저장된 대량의 데이터로부터 사용자가 원하는 정보를 효과적으로 검색하는 기능이다.
- 그 동안 컴퓨터에 저장된 데이터를 관리하기 위하여 다양한 형태의 파일 시스템이 사용되어 왔으나, 1970년대부터 대량의 데이터를 체계적으로 관리하기 위한 데이터베이스 관리시스템(DBMS)이 널리 사용되고 있다.

- DBMS는 컴퓨터에 저장된 대량의 데이터를 체계적으로 관리하고 사용자가 원하는 정보를 효과적으로 검색하기 위한 시스템 소프트웨어를 의미한다. 특히 DBMS는 기업이나 기관에서 사용되는 데이터를 중앙 집중식으로 관리하여 다수의 사용자들이 정보를 공유할 수 있는 환경을 제공한다.
- DBMS에 의해 관리되는 데이터의 집합을 '데이터베이스'라고 한다.
- 일반 사용자들이 데이터베이스에 저장된 데이터를 검색하고 조작하기 위한 언어를 데이터베이스 언어라고 한다. 가장 널리 사용되는 데이터베이스 언어는 미국표준연구소(ANSI:American National Standard Institute)와 국제표준기구(ISO:International Standard Organization)에서 관계형 데이터베이스 표준언어로 채택한 SQL(Structured Query Language)언어이다.

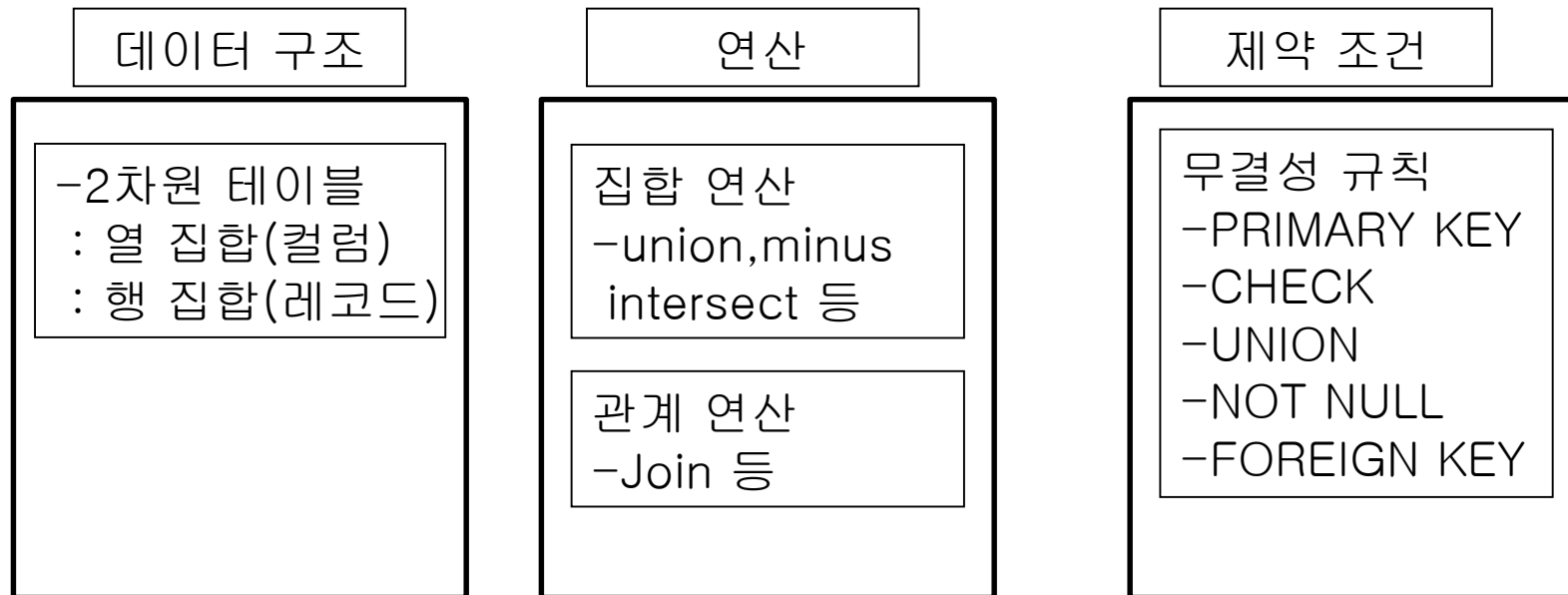
- 데이터베이스는 대학, 기업, 관공서등과 같은 조직에서 사용되는 데이터를 중앙 집중적으로 관리하여 다수의 사용자들이 다양한 응용 분야에서 동일한 데이터를 공유하기 위한 목적으로 사용한다.

따라서 다음과 같이 정의할 수 있다.

특정 조직에서 운영되는 여러 응용 시스템에서 공유하기 위해 통합, 저장된 운영 및 의사결정 데이터의 집합.



- 일반적으로 데이터 모델은 데이터구조(data structure), 연산(operation), 제약조건(constraint)의 3요소로 설명된다.
- 관계형 데이터 모델은 2차원 테이블 형태의 데이터 구조를 갖는다.



- 관계형 데이터베이스(Relational Database)는 일련의 정형화된 데이터 항목들의 2차원 테이블 형태의 집합체로서, 데이터베이스를 만들거나 이용하기가 비교적 쉽지만, 무엇보다도 확장이 용이하다는 장점을 가지고 있다. 처음 데이터베이스를 만든 후 관련되는 응용 프로그램들을 변경하지 않고도, 새로운 데이터 항목을 데이터베이스에 추가할 수 있다는 장점이 있다.

성명(컬럼)	주소(컬럼)	전화번호(컬럼)
성명	주소	전화번호
홍길동	서울	02-111-2222
이순신	부산	02-333-1245
강감찬	대구	02-987-1234

고객 정보(로우)

고객 테이블



- SQL 언어는 오늘날 대부분의 데이터베이스 응용 분야에서 관계형 데이터베이스 제품이 사용 되는데 가장 핵심적인 역할을 담당하고 있다.
- 오늘날 대부분의 상용 관계형 데이터베이스에서 데이터베이스 표준 언어로 사용되고 있다.

년도	설명
1973	SQUARE(Structured Queries As Relational Express)
1974	SEQUEL(Structured English QUery Language )
1980	SQL(Structured Query Language)로 개명
1986	SQL-86, 최초의 SQL 표준안
1988	ANSI, ISO 국제 표준 인정
1989	SQL-1(SQL/89) 표준안 제정
1992	SQL-2(SQL/92) 표준안 제정
1999	SQL-3(SQL/99) 표준안 제정

종류	명령문
DQL(Data Query Language) : 질의어	select (데이터 검색시 사용)
DML(Data Manipulation Language) : 데이터 조작어	insert (데이터 입력) update (데이터 수정) delete (데이터 삭제)
DDL(Data Definition Language) : 데이터 정의어	create (객체 생성) alter (객체 수정) drop (객체 삭제) rename (객체이름 변경) truncate (객체저장공간 삭제)
TCL(Transaction Control Language ) : 트랜잭션 처리어	commit (트랜잭션 확정) rollback (트랜잭션 취소) savepoint (트랜잭션내의 체크포인트)
DCL(Data Control Language ) : 데이터 제어어	grant (권한 부여) revoke (권한 취소)

- SQL문은 대소문자를 구별하지 않는다.  
기본적으로 테이블이나 컬럼 이름은 대문자로 만들어진다.
- SQL문은 연산자와 함수를 제공한다.
- SQL문은 여러 줄에 걸쳐서 작성 가능하고 반드시 ;(세미콜론)으로 끝난다.
- SQL문은 자바언어와 마찬가지로 키워드가 있으며 대문자 사용을 권장한다.

## ○ 예약어

```
select keyword  
from v$reserved_words  
order by keyword;
```

## ○ 연산자

산술 연산자: +, -, \*, /

연결 연산자: ||

비교 연산자: =, >, >=, <, <=, !=

논리 연산자: and, or, not

Set 연산자: union, union all, intersect, minus

계층형 쿼리연산자: prior, connect by 등

## ○ 함수

- 문자함수
- 숫자함수
- 날짜함수 등

## ○ 의사코드

**ROWNUM:** 쿼리 결과로 나오게 되는 각각의 행들에 대한 순서 값을 가리킨다.  
주로 특정 개수나 그 이하의 행을 선택할 때 사용된다.

예> emp 테이블에서 10건의 데이터만 조회?

```
select *
from emp
where ROWNUM < 10;
```

**ROWID:** 테이블에 저장된 각각의 행들이 저장된 주소값을 가리킨다.  
즉 행들의 고유 주소값이다.

AAAMg3AAFAAAABUAAA

Object id    file id    Block id    Block 내의 row



**Thank you**

---