

# 6장 SQL 문의 기본

## SELECT

- 정렬 수행 시 정렬키가 같은 값의 행이 복수개 존재한다면 그 행들의 순서는 일정하지 않다.  
행의 순서를 확실히 같게 하려면 행의 정렬키를 Unique key 로 정해야 한다.

```
mysql> SELECT * FROM city WHERE countrycode='KOR' ORDER BY district;  
  
# 이렇게 하면 일정한 순서로 나열 된다.  
mysql> SELECT * FROM city WHERE countrycode='KOR' ORDER BY district, name;
```

## 함수

- 복수행이나 행의 값에 대해 집계를 수행하는 함수 → 집계함수
- 단일 행의 값에 대해 조작이나 계산을 수행하는 함수

## 집계 함수

- COUNT, SUM, AVG, MAX, MIN
- 기본적으로 NULL 을 빼고 집계
- COUNT 함수만은 NULL 을 포함한 전체 행을 집계한다.

```
mysql> SELECT district, count(*) FROM city WHERE countrycode='KOR' and count(*)=4 GROUP BY district;
```

▼ 위의 쿼리는 사용할 수 없다.

집계 함수는 SELECT, ORDER BY, HAVING 안에서만 사용할 수 있다.

WHERE 에서 집계 함수를 사용할 수 없기 때문에 이 경우는 HAVING 으로 지정해야 한다.

▼ 위를 맞게 고치면?

```
mysql> SELECT district, count(*) FROM city WHERE countrycode='KOR' GROUP BY district HAVING count(*)=4;
```

## 명령어 순서

1. SELECT
2. FROM
3. WHERE
4. GROUP BY
5. HAVING
6. ORDER BY

## INSERT

- 기본값으로 DEFAULT 키워드를 사용할 수 있다.
- `INSERT INTO 테이블1 SELECT * FROM 테이블2` 와 같은 형태도 사용 가능
- Bulk insert query
- postgresSQL - copy command

## SQL 표준어와 비 표준어의 차이

처음에 SQL 이 등장했을 때 표준이 없어서 각 벤더가 독자적으로 구현한 비 표준어가 존재한다.

- 준비된 데이터형, 함수의 기능이나 범위가 다르다.
- 특정 데이터형, 비교 연산자에서 NULL 취급이 다르다  
예를 들면 VARCHAR2 형에서는 빈 문자형도 NULL 로 취급하는 등
- 내부결합 (INNER JOIN) 이나 외부 결합 (OUTER JOIN) 에 SQL 표준 이외의 오래된 표기나 벤더 독자 표기가 있다.
- 데이터베이스 특유의 사용자 관리를 하지 않고 OS 사용자와 연계할 수 있다.
- 스토어드 루틴(프로시저, 함수)나 트리거 유무에도 차이가 있다.
- 때로는 기술 언어에도 차이가 있다 (PL/SQL, T-SQL, SQL/PSM)
- 비교적 새로운 SQL의 표준 기능의 유무 (Window 함수, SQL/MED 등)

## 뷰

- 테이블과 동일하지만 테이블과 같은 데이터를 가지고 있지 않으며 테이블에 대한 SELECT 를 가지고 있는 것을 뷰라고 한다. 여기서 말하는 데이터란 기본 테이블에서 일정 시가나마다 반영되는 구조를 가지고 있는 것

- 이점
  1. 복잡한 SELECT 문을 일일이 매번 기술할 필요가 없다.
  2. 필요한 열과 행만 사용자에게 보여줄 수 있고, 갱신 시에도 뷰 정의에 따른 갱신으로 한정할 수 있다.

위의 두가지를 기억장치의 용량을 사용하지 않고 실현할 수 있다.
- 단, 뷰로 입력하거나 갱신할 때는 제한이 따른다.

## JOIN

