

SELECT 문법

SELECT는 한 개 또는 여러 개의 테이블의 열을 검색할 수 있습니다. SELECT의 일반적인 형태는 다음과 같습니다.

SELECT Syntax

```
1  SELECT [ ALL | DISTINCT [ ON ( expression [, ...] ) ] ]
2      [ * | expression [ [ AS ] output_name ] [, ...] ]
3      [ FROM from_item [, ...] ]
4      [ WHERE condition ]
5      [ GROUP BY expression [, ...] ]
6      [ HAVING condition [, ...] ]
7      [ WINDOW window_name AS ( window_definition ) [, ...] ]
8      [ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select ]
9      [ ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST | LAST } ] [, ...] ]
10     [ LIMIT { count | ALL } ]
11     [ OFFSET start [ ROW | ROWS ] ]
```

SELECT절에서 쓰일 수 있는 Option은 다음과 같습니다.

Option	기능
SELECT ALL	중복되는 열까지 결과값으로 출력해 주는 역할
SELECT DISTINCT ON	중복되는 열들을 결과값에서 제거하는 역할
AS	alias의 약자로 선택한 column의 이름을 AS를 사용하면 새로운 이름으로 지정하여 출력할 수 있습니다.
FROM	한 개 또는 그 이상의 테이블을 지정할 수 있습니다.
WHERE	FROM절의 테이블 내에서 조건을 만족하는 행을 검사합니다.
GROUP BY	동일한 값을 가진 데이터를 집계해서 조회하고자 할 때 사용합니다. 예를 들어 각 부서의 평균연령, 부서의 인원이 몇 명인지를 집계해서 다른 부서와 비교할 수 있습니다.
HAVING	절은그룹을 가지고 조건을 비교할 때 사용합니다. GROUP BY와 함께 쓰입니다. 예를들어 주어진 조건을 만족하지 않은 그룹을 제거할수 있습니다.
UNION	열의 개수가 동일하고 타입이 동일한 2개의 집합들의 결과값을 결합하여 반환합니다.
INTERSECT	교집합과 같은 원리로 두 집합모두에 있는 열들을 반환합니다.
EXCEPT	차집합과 같은 원리로 두 집합모두에 있는 열들을 반환합니다.
ORDER BY	파일을 정렬하는데 ASC와 DESC를 사용하여 오름차순과 내림차순을 설정할 수 있습니다.
LIMIT	쿼리결과 의 개수를 제한할 수 있습니다.
OFFSET	시작 행을 반환하기 전에 많은 행을 건너뛰는 역할을 합니다.

SELECT 예제

Weather_reports 테이블에서 time, report, location컬럼을 추출합니다.

이때 location컬럼에서 중복되는 컬럼은 제거합니다. 마지막으로 행을 정렬할 때 location은 오름차순, time은 내림차순으로 정렬합니다.

```
1  SELECT DISTINCT ON (location) location, time, report
2      FROM weather_reports
3      ORDER BY location, time DESC;
```

Student 테이블에서 student_no가 5인 사람의 모든 정보를 출력합니다.

```
1  SELECT *
2      FROM student
3      WHERE student_no= 5;
```

films테이블에서 kind컬럼이 같은 것끼리 그룹을 짓습니다. 그리고 len컬럼의 총합을 total로 지정합니다. 그리고 sum(len)의 값이 5시간보다 작은 것들을 선택해서 출력합니다.

```
1  SELECT kind, sum(len) AS total
2      FROM films
3      GROUP BY kind
4      HAVING sum(len) < interval '5 hours';
```

Distributors테이블에서 name이 W로 시작하는 사람을 추출하고 actors테이블에서 name이 W로 시작하는 사람을 추출하여 함께 출력합니다.

```
1 SELECT distributors.name
2 FROM distributors
3 WHERE distributors.name LIKE 'W%'
4 UNION
5 SELECT actors.name
6 FROM actors
7 WHERE actors.name LIKE 'W%';
```