

◎ 데이터 제한 및 정렬

-- 선택되는 행 제한

- WHERE 절을 사용하여 query에서 반환되는 행을 제한할 수 있습니다. WHERE 절은 충족되어야 하는 조건을 포함하며 FROM 절 바로 뒤에 나옵니다. 조건이 참인 경우 해당 조건을 충족하는 행이 반환됩니다.

- 이 구문에서 다음이 적용됩니다.

WHERE	=====	조건을 충족하는 행으로 query를 제한합니다.
condition	=====	열 이름 , 표현식, 상수 및 비교 연산자로 구성됩니다. 조건은 하나 이상의 표현식과 논리(부울) 연산자의 조합을 지정하고 TRUE, FALSE 또는 UNKNOWN의 값을 반환합니다.

- WHERE 절은 열의 값, 리터럴, 산술식 또는 함수를 비교할 수 있으며 다음 세 가지 요소로 구성됩니다.

- 열 이름
- 비교 조건
- 열 이름 , 상수 또는 값 리스트

비교 연산자

연산자	의미
=	같음
>	보다 큼
>=	보다 크거나 같음
<	보다 작음
<=	보다 작거나 같음
<>	같지 않음
BETWEEN ...AND...	두 값 사이(경계값 포함)
IN(set)	값 리스트 중 일치하는 값 검색
LIKE	일치하는 문자 패턴 검색
IS NULL	null 값인지 여부

- 비교 연산자는 특정 표현식을 다른 값이나 표현식과 비교하는 조건에서 사용됩니다. WHERE 절에서 다음과 같은 형식으로 사용됩니다.

구문

... WHERE expr operator value

예제

... WHERE hire_date = '01-JAN-95'

... WHERE salary >= 6000

... WHERE last_name = 'Smith'

- WHERE 절에서는 alias를 사용할 수 없습니다.

참고: != 및 ^= 기호도 같지 않음 조건을 나타냅니다.

-- 논리 연산자를 사용한 제한(조건 정의)

논리 연산자를 사용하여 조건 정의

연산자	의미
AND	구성 요소 조건이 모두 참인 경우 TRUE를 반환합니다.
OR	구성 요소 조건 중 하나가 참인 경우 TRUE를 반환합니다.
NOT	조건이 거짓인 경우 TRUE를 반환합니다.

: 논리 조건은 두 구성 요소 조건의 결과를 결합하여 해당 조건을 기반으로 단일 결과를 산출하거나, 단일 조건의 결과를 뒤바꿉니다. 조건의 전체 결과가 참인 경우에만 행이 반환됩니다.

SQL에서는 다음 세 가지 논리 연산자가 제공됩니다.

- AND
- OR
- NOT

지금까지의 모든 예제는 WHERE 절에서 하나의 조건만 지정했습니다. AND 및 OR 연산자를 사용하여 하나의 WHERE 절에서 여러 조건을 사용할 수 있습니다.

- AND 연산자 사용

: 레코드가 선택되려면 두 구성 요소 조건이 모두 참이어야 합니다. 모든 문자 검색에서는 대소문자가 구분되어야 하고 문자열은 따옴표로 묶어야 합니다.

: AND 진리표

ANDTRUE	TRUE	FALSE	NULL
TRUE	TRUE	FALSE	NULL
FALSE	FALSE	FALSE	FALSE
NULL	NULL	FALSE	NULL

[예제]

- OR 연산자 사용

: 레코드가 선택되려면 두 구성요소 조건 중 하나가 참이어야 합니다.

: OR 진리표

OR	TRUE	FALSE	NULL
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	NULL
NULL	TRUE	NULL	NULL

- NOT 연산자 사용

NOT	TRUE	FALSE	NULL
	FALSE	TRUE	NULL

[예제]

참고 : NOT연산자는 BETWEEN, LIKE, NULL 등의 다른 SQL연산자와 함께 사용 가능

● 표현식 우선 순위 규칙

우선 순위 규칙

연산자	의미
1	산술 연산자
2	연결 연산자
3	비교 조건
4	IS [NOT] NULL, LIKE, [NOT] IN
5	[NOT] BETWEEN
6	같지 않음
7	NOT 논리 조건
8	AND 논리 조건
9	OR 논리 조건

괄호를 사용하여 우선 순위 규칙을 재정의할 수 있습니다.

-- 행 정렬

: ORDER BY 절 사용하여 검색된 행을 정렬합니다.

: Query 결과에 반환되는 행의 순서는 정의되어 있지 않습니다. ORDER BY 절을 사용하여 행을 정렬할 수 있습니다. 그러나 SQL 문에서 ORDER BY 절을 사용하는 경우 그 뒤에 다른 절을 사용할 수 없습니다. 또한 표현식, alias 또는 열 위치를 정렬 조건으로 지정할 수 있습니다.

구문

```
SELECT      expr
FROM        table
[WHERE      condition(s)]
[ORDER BY   {column, expr, numeric_position} [ASC|DESC]];
```

이 구문에서 다음이 적용됩니다.

ORDER BY 검색된 행이 표시되는 순서를 지정합니다.
ASC 오름차순으로 행을 정렬합니다(기본 순서).

DESC 내림차순으로 행을 정렬합니다.

ORDER BY 절을 사용하지 않는 경우에는 정렬 순서가 정의되지 않으며 Oracle 서버는 동일한 query에 대해 동일한 순서로 행을 두 번 패치(fetch)하지 못할 수 있습니다. 특정 순서로 행을 표시하려면 ORDER BY 절을 사용하십시오.

참고: NULLS FIRST 또는 NULLS LAST 키워드를 사용하여 null 값을 포함하는 반환된 행이 정렬 순서상 맨 처음 나타나거나 마지막에 나타나도록 지정할 수 있습니다.

[내림차순으로 정렬]

```
SQL> select empno, ename, sal
  2  from emp
  3  order by sal desc;
```

[열 ALIAS를 기준으로 정렬]

```
SQL> select ename, sal*12 fullsal
  2  from emp
  3  order by fullsal desc;
```

[열의 숫자 위치를 사용하여 정렬]

```
SQL> ed
file afiedt.buf(0)가 기록되었습니다

  1  select ename, sal*12 fullsal
  2  from emp
  3* order by 2 desc
SQL> /
```

[여러 열을 기준으로 정렬]

```
SQL> select ename, sal
  2  from emp
  3  order by sal desc, ename desc;
```

