문법:

SELECT [COLUMN], [COLUMN]

전체일 경우, SELECT \*

FROM [TABLE]

ORDER BY [COLUMN] [ORDER TYPE]

정렬방식에선 ASC, DESC 존재

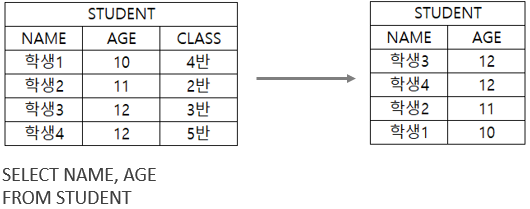
ASC: 오름차순

DESC: 내림차순

WHERE [CONDITIONS]

컬럼 값과 VALUE 를 비교함.

자바의 비교연산과 동일한 방법



TO\_CHAR(DATE 타입의 날짜 값 혹은 컬럼, ‘날짜 포멧‘)

* + 날짜타입의 데이터를 문자로 변경한다.

TO\_DATE(문자열타입의 날짜 값 혹은 컬럼, ‘날짜 포멧‘)

* + 문자 타입의 날짜 데이터를 날짜 타입으로 변경한다.

LPAD(숫자 타입의 값 또는 컬럼, 채울 자리수, 빈 공간에 채울 문자)

* + 숫자 왼쪽 공간에 문자를 채운다.
  + LPAD(1, 6, ‘0’) 🡺 000001

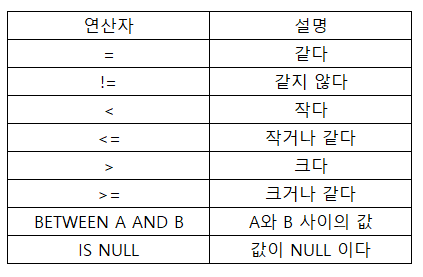
NVL(NULL일 가능성이 있는 컬럼, 기본값)

* + 값 또는 컬럼이 NULL이라면 기본값으로 표현한다.

CASE WHEN ELSE END

* + Java의 if 와 유사함.

**SQL연산자**



IN, LIKE, NOT

IN : 하나의 컬럼에 여러 개의 값을 검색함

* 최대 1000개까지만 들어갈 수 있음
* 내부에서 연산은 불가능함.

LIKE : 하나의 컬럼에 “포함된” 값을 검색함.

* WILD CARD 가 사용됨.
* %, \_

**WHERE절은 여러 개의 조건을 지원한다.**

AND, OR 를 사용해 여러 개의 조건을 사용할 수 있다.

WHERE [CONDITION1]

AND [CONDITION2]

AND [CONDITION3]

**AND와 OR 가 동시에 사용될 때 AND 가 우선순위를 가진다.**

WHERE [CONDITION1]

AND [CONDITION2]

AND [CONDITION3]

OR [CONDITION4]

AND [CONDITION5]

🡺 CONDITION1 AND CONDITION2

🡺 CONDITION2 AND CONDIDTION3

🡺 CONDITION3 OR (CONDITION4 AND CONDITION5)

CONDITION4 AND CONDITION5 가 우선처리됨.