
SQL - SELECT

정종욱 jongukjeong@gmail.com

SELECT

- DESC 명령어로 컬럼을 확인하자!

SCOTT>DESC emp ;

Name	Null?	Type
-----	-----	-----
EMPNO	NOT NULL	NUMBER(4)
ENAME		VARCHAR2(10)
JOB		VARCHAR2(9)
MGR		NUMBER(4)
HIREDATE		DATE
SAL		NUMBER(7,2)
COMM		NUMBER(7,2)
DEPTNO		NUMBER(2)

SELECT

- SELECT : 데이터 조회하기

문법 : SELECT [컬럼명 또는 표현식] FROM [테이블명 , 뷰 명] ;

- 모든 컬럼 조회하기

```
SCOTT>SELECT * FROM emp ;
```

```
SCOTT>SELECT *  
2 FROM emp ;
```

```
SCOTT>SELECT *  
2 FR  
3 OM emp;  
FR  
*
```

```
ERROR at line 2:  
ORA-00923: FROM keyword not found where expected
```

키워드는 줄 바꾸면 안됨!!

SELECT

1. 대소문자 구분을 하지 않아도 실행되지만 원래는 다른 문장이다!

2. 한 줄 또는 여러 줄에 걸쳐 작성해도 되며 마지막은 ; (세미콜론)으로 끝맺어야만 한다.

3. SQL 키워드는 분리해서는 안 된다.

SQL 키워드라 함은 SELECT , FROM , WHERE 등과 같이 SQL 에서 사용하는 미리 정해놓은 단어를 말한다.

SELECT

- 화면에 보기 좋게 출력하는 팁

```
SCOTT> COL empno FOR 9999 ;
```

숫자일 경우

Empno라는 컬럼의 길이를 숫자 4자리까지 들어가게 설정

```
SCOTT> COL ename FOR a8 ;
```

문자일 경우

Ename의 컬럼 길이를 8바이트까지 들어가도록 설정

SELECT

- 화면에 보기 좋게 출력하는 팁

```
SCOTT> SET line 200 ;
```

한 줄에 최대 200글자(byte)까지 나오도록 설정

```
SCOTT> SET pagesize 50 ;
```

한 화면에 최대 50줄까지 나오도록 설정

SELECT

```
SCOTT>SELECT empno , ename  
2 FROM emp ;
```

EMPNO	ENAME
7369	SMITH
7499	ALLEN
7521	WARD
7566	JONES
7654	MARTIN
7698	BLAKE
(이하 생략)	

14 rows selected.

원하는 컬럼만
조회하는 방법

SELECT

- 표현식을 사용하여 출력하기 (리터럴 상수, Literal)
 - 원래 컬럼 이외에 출력하기를 원하는 내용

```
SCOTT>SELECT name , '교수님~배고파요~'  
2      FROM professor ;
```

NAME	'교수님~배고파요'
------	------------

조인형	교수님~배고파요~
-----	-----------

박승곤	교수님~배고파요~
-----	-----------

주승재	교수님~배고파요~
-----	-----------

(이하 생략)

SELECT

- 칼럼 별칭 사용하여 출력하기 - 1

```
SCOTT>SELECT studno, name  
2 FROM student ;
```

STUDNO	NAME
9411	서진수
9412	서재수
9413	이미경
9414	김재수
(이하 생략)	

- 칼럼 별칭 사용 하기 전
소문자가 대문자로 출력됨

SELECT

- 칼럼 별칭 사용하여 출력하기 - 2

SCOTT>SELECT studno "학번", name AS "이름", profno "지도교수번호"
2 FROM student ;

학번	이름	지도교수번호
----	----	--------

9411	서진수	1001
9412	서재수	2001
9413	이미경	3002

(이하 생략)

쌍따옴표는 공백, 특수문자, 대소문자의 구분이 필요한 경우 꼭 사용

SELECT

- Alias 연습문제 1

emp 테이블을 사용하여 empno 를 사원번호 , ename 을 사원명 , job 을 직업으로 별명을 설정하여 출력하세요.

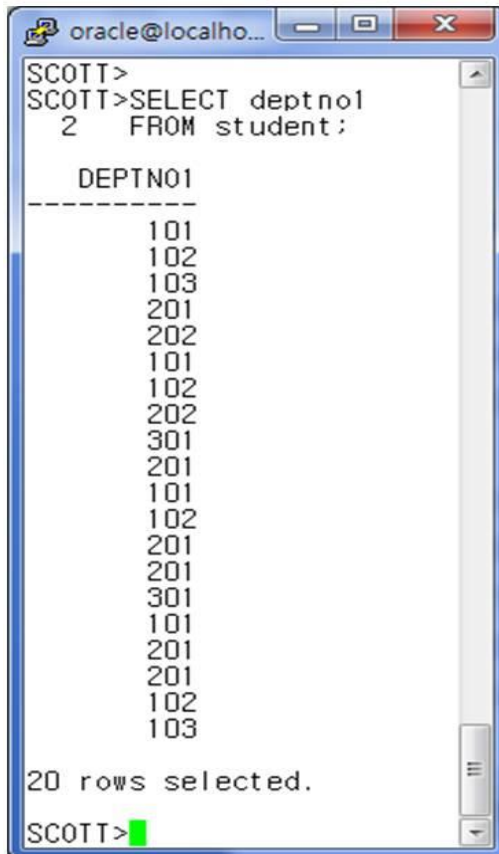
- Alias 연습문제 2

dept 테이블을 사용하여 deptno 를 부서# , dname 를 부서명 , loc 를 위치로 별명을 설정하여 출력하세요.

SELECT

(5) DISTINCT - 중복된 값을 제거하고 출력하기

DISTINCT 사용 전



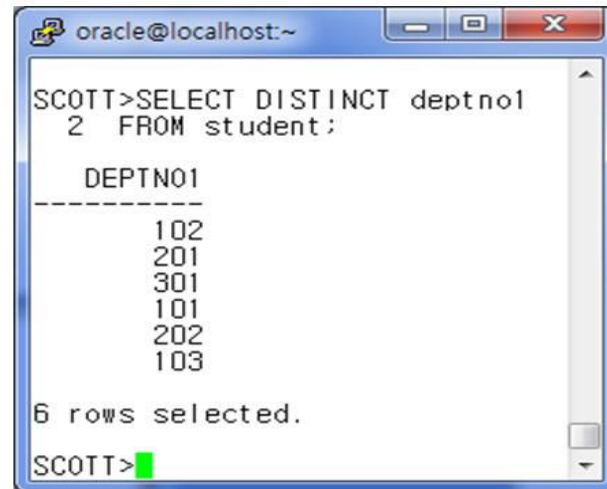
```
SCOTT>
SCOTT>SELECT deptno1
2 FROM student;

DEPTNO1
-----
101
102
103
201
202
101
102
202
301
201
101
102
201
201
301
101
201
201
102
103

20 rows selected.

SCOTT>
```

DISTINCT 사용 후



```
SCOTT>SELECT DISTINCT deptno1
2 FROM student;

DEPTNO1
-----
102
201
301
101
202
103

6 rows selected.

SCOTT>
```

SELECT

DISTINCT 사용 안 함

```
oracle@localhost:~  
SCOTT>  
SCOTT>SELECT deptno ,job  
2 FROM emp ;
```

DEPTNO	JOB
20	CLERK
30	SALESMAN
30	SALESMAN
20	MANAGER
30	SALESMAN
30	MANAGER
10	MANAGER
20	ANALYST
10	PRESIDENT
30	SALESMAN
20	CLERK
30	CLERK
20	ANALYST
10	CLERK

14 rows selected.

1

DISTINCT 사용 함

```
oracle@localhost:~  
SCOTT>  
SCOTT>SELECT DISTINCT deptno  
2 FROM emp ;
```

DEPTNO
30
20
10

2

두 컬럼 DISTINCT 사용 함

```
oracle@localhost:~  
SCOTT>  
SCOTT>SELECT DISTINCT deptno , job  
2 FROM emp ;
```

DEPTNO	JOB
20	CLERK
30	SALESMAN
20	MANAGER
30	CLERK
10	PRESIDENT
30	MANAGER
10	CLERK
10	MANAGER
20	ANALYST

9 rows selected.

3

SELECT

(6) 연결(합성) 연산자 (Concatenation)로 칼럼을 붙여서 출력하기

연결연산자 미 사용

```
SCOTT>SELECT name , position  
2 FROM professor ;
```

NAME	POSITION
-----	-----
조인형	정교수
박승곤	조교수
송도권	전임강사
양선희	전임강사
김영조	조교수
주승재	정교수
김도형	정교수
나한열	조교수
김현정	전임강사
심슨	정교수
최슬기	조교수
박원범	조교수
차범철	전임강사
바비	정교수
전민	전임강사
허은	조교수

16 rows selected.

연결연산자 사용

```
SCOTT>SELECT name || position  
2 FROM professor ;
```

NAME POSITION

조인형정교수
박승곤조교수
송도권전임강사
양선희전임강사
김영조조교수
주승재정교수
김도형정교수
나한열조교수
김현정전임강사
심슨정교수
최슬기조교수
박원범조교수
차범철전임강사
바비정교수
전민전임강사
허은조교수

16 rows selected.

SELECT

```
SCOTT>SELECT name || '님은 ' || position || '입니다'  
2 FROM professor ;
```

NAME||'님은'||POSITION||'입니다'

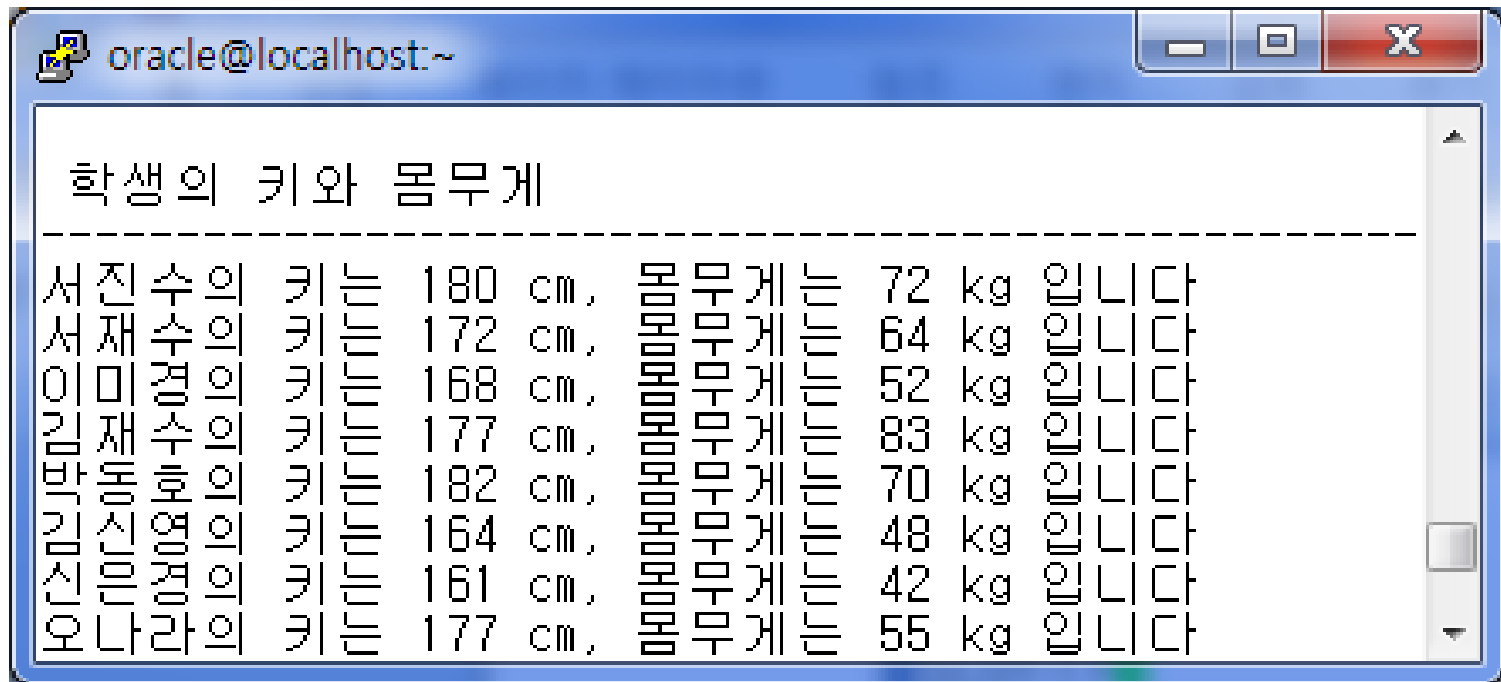
NAME	POSITION
조인형	정교수
박승곤	조교수
송도권	전임강사
양선희	전임강사
김영조	조교수
주승재	정교수
김도형	정교수
나한열	조교수
김현정	전임강사
심슨	정교수

(이하 생략)

리터럴 문자를 사용한 경우

SELECT

연결 연산자 문제 1 : 학생 테이블(student)을 사용하여 모든 학생들이 '서진수 의 키는 180 cm, 몸무게는 55 kg 입니다' 와 같은 형식으로 출력되도록 리터럴 문자를 추가하고 , 컬럼이름은 "학생의 키와 몸무게"라는 별명으로 출력해 보세요.



학생의 키와 몸무게									
서진수	의	키	는	180	cm,	몸	무	게	는
서재수	의	키	는	172	cm,	몸	무	게	는
이미경	의	키	는	168	cm,	몸	무	게	는
김재수	의	키	는	177	cm,	몸	무	게	는
박동호	의	키	는	182	cm,	몸	무	게	는
김신영	의	키	는	164	cm,	몸	무	게	는
신은정	의	키	는	161	cm,	몸	무	게	는
오나라	의	키	는	177	cm,	몸	무	게	는

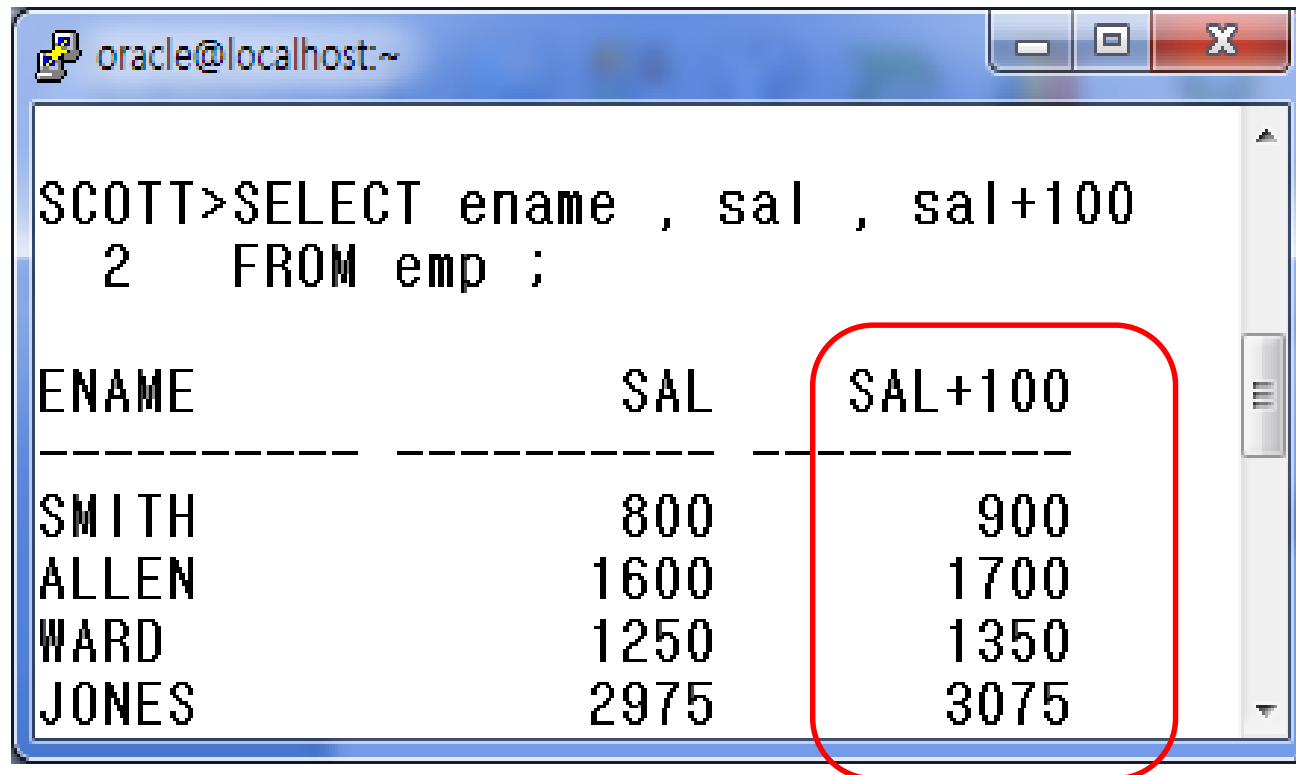
SELECT

연결 연산자 문제 2: 홍길동 (교수) , 홍길동 '교수' 이렇게 나오도록 출력해보세요

[illegible]

SELECT

(7) 산술 연산자 사용하기 : +, -, *, /

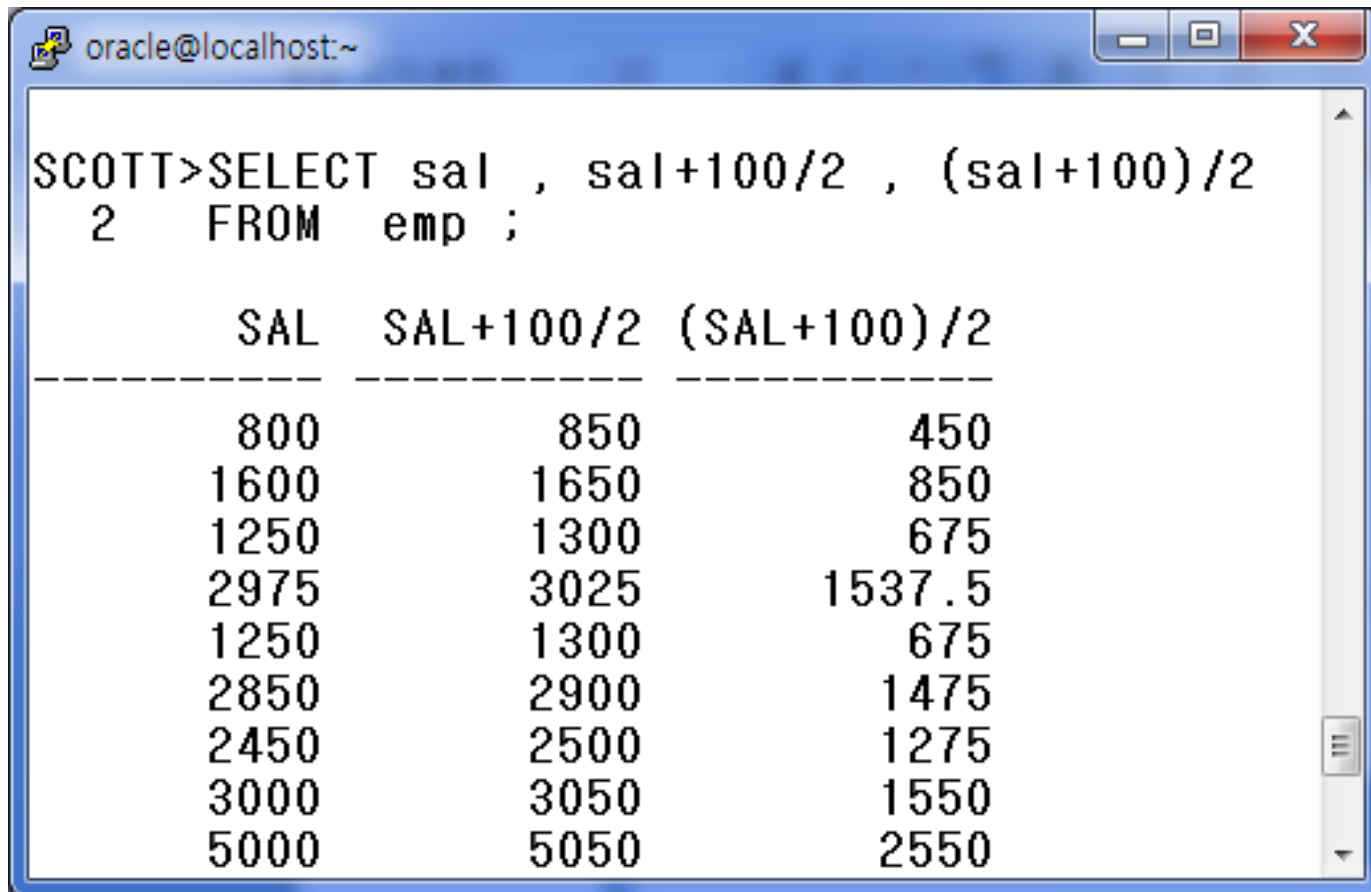


The screenshot shows a terminal window titled 'oracle@localhost:~'. The prompt is 'SCOTT>' and the command entered is 'SELECT ename , sal , sal+100 FROM emp ;'. The output is a table with three columns: 'ENAME', 'SAL', and 'SAL+100'. The 'SAL+100' column is highlighted with a red rounded rectangle. The data rows are SMITH (800, 900), ALLEN (1600, 1700), WARD (1250, 1350), and JONES (2975, 3075).

ENAME	SAL	SAL+100
SMITH	800	900
ALLEN	1600	1700
WARD	1250	1350
JONES	2975	3075

SELECT

- 산술 연산자 사용시 우선순위 주의 할 것!



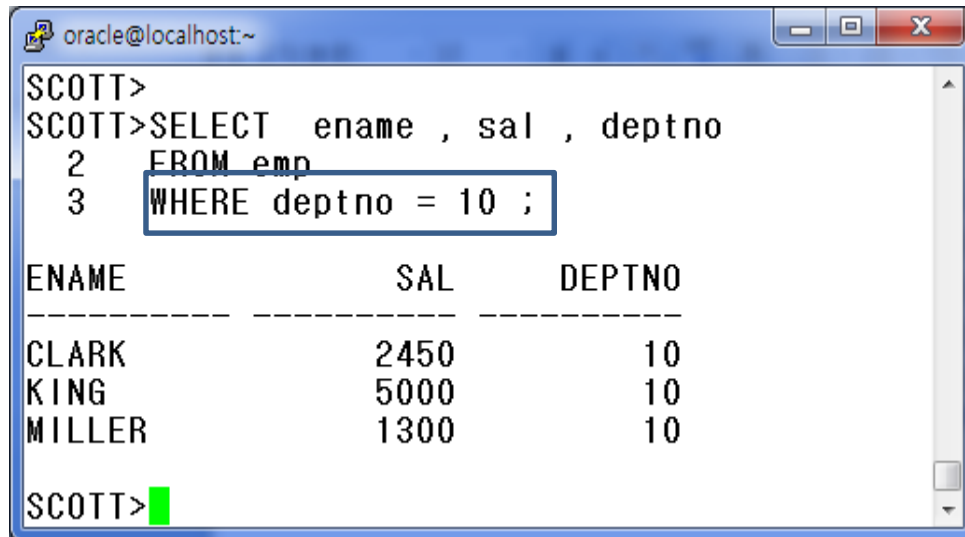
The screenshot shows a terminal window with the prompt 'oracle@localhost:~'. The user has entered the SQL command: `SCOTT>SELECT sal , sal+100/2 , (sal+100)/2`. The output displays a table with three columns: SAL, SAL+100/2, and (SAL+100)/2. The results are as follows:

SAL	SAL+100/2	(SAL+100)/2
800	850	450
1600	1650	850
1250	1300	675
2975	3025	1537.5
1250	1300	675
2850	2900	1475
2450	2500	1275
3000	3050	1550
5000	5050	2550

SELECT

(8) WHERE 절을 활용하여 원하는 조건만 조회하기

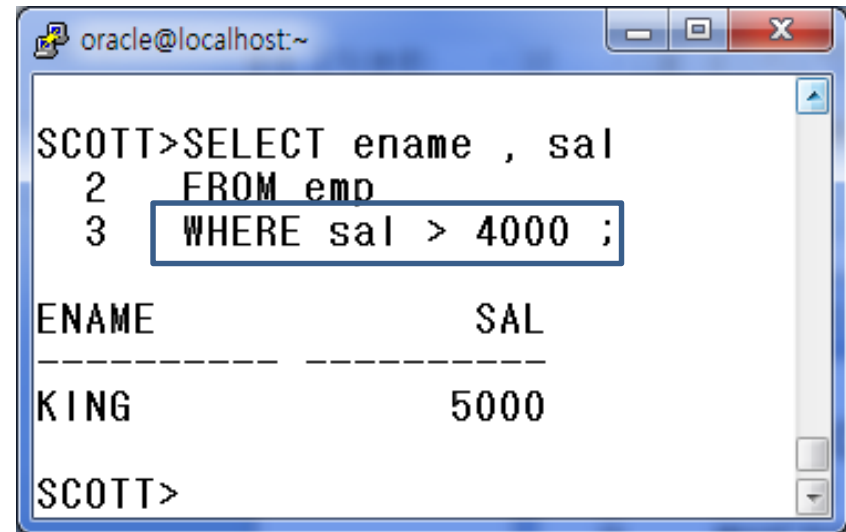
```
SQL>SELECT [ Column or Expression ]  
2     FROM [ Table or View ]  
3     WHERE 원하는 조건 ;
```



```
oracle@localhost:~  
SCOTT>  
SCOTT>SELECT  ename , sal , deptno  
2     FROM emp  
3     WHERE deptno = 10 ;
```

ENAME	SAL	DEPTNO
CLARK	2450	10
KING	5000	10
MILLER	1300	10

```
SCOTT>
```



```
oracle@localhost:~  
SCOTT>SELECT  ename , sal  
2     FROM emp  
3     WHERE sal > 4000 ;
```

ENAME	SAL
KING	5000

```
SCOTT>
```

SELECT

- 문자열 조회할 때 주의 사항

```
oracle@localhost:~  
SCOTT>SELECT ename , empno , sal  
2 FROM emp  
3 WHERE ename = 'SCOTT';  
  
ENAME          EMPNO          SAL  
-----  
SCOTT          7788          3000  
  
SCOTT>
```

```
oracle@localhost:~  
SCOTT>SELECT ename , empno , sal  
2 FROM emp  
3 WHERE ename = 'scott';  
  
no rows selected
```

WHERE 절의 문자는
대소문자 구분 합니다!
홀따옴표로 묶으세요!

SELECT

- 날짜 조회할 때 주의 사항

```
oracle@localhost:~  
SCOTT>SELECT name , pay , hiredate  
2 FROM professor  
3 WHERE hiredate = 23-MAY-01;  
WHERE hiredate = 23-MAY-01  
*  
ERROR at line 3:  
ORA-00904: "MAY": invalid identifier
```

- 홀따옴표로 묶으세요
- 대소문자 구분 없습니다.

```
oracle@localhost:~  
SCOTT>SELECT name , pay , hiredate  
2 FROM professor  
3 WHERE hiredate = '23-MAY-01';  
  
NAME                PAY HIREDATE  
-----  
허은                290 23-MAY-01
```

```
oracle@localhost:~  
SCOTT>SELECT name , pay , hiredate  
2 FROM professor  
3 WHERE hiredate = '23-may-01';  
  
NAME                PAY HIREDATE  
-----  
허은                290 23-MAY-01
```

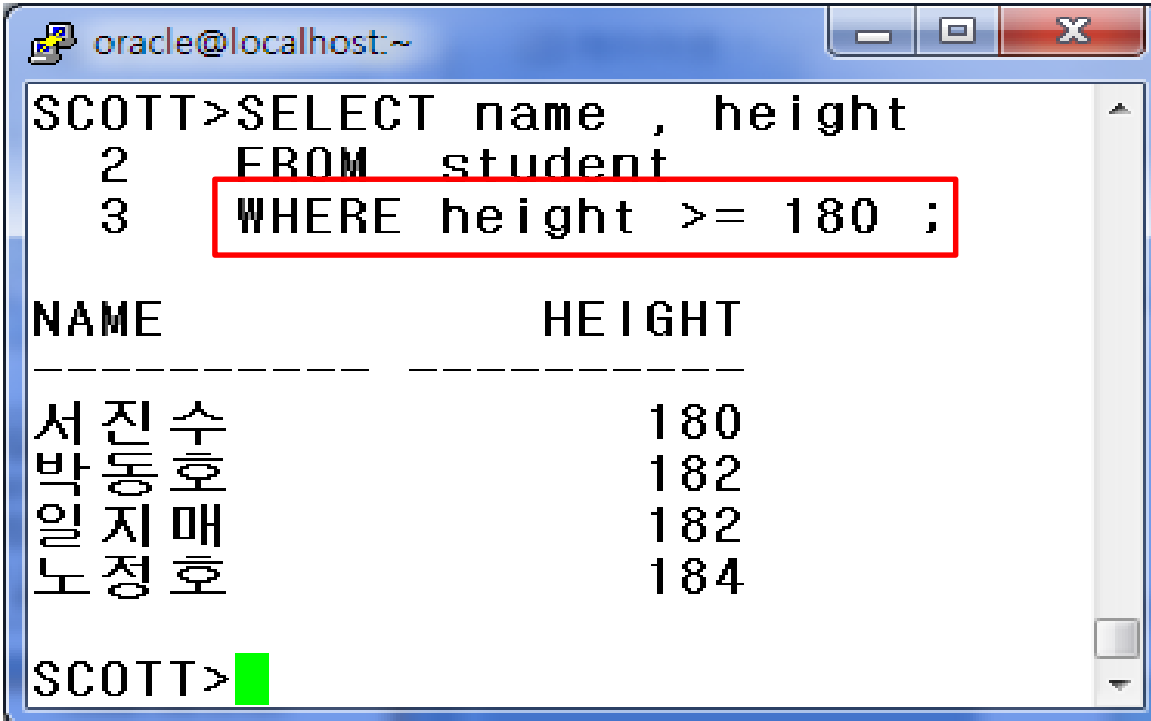
SELECT

(9) 다양한 연산자를 활용하는 방법

연산자 종류	설 명
=	같은 조건을 검색
!= , <>	같지 않은 조건을 검색
>	큰 조건을 검색
>=	크거나 같은 조건을 검색
<	작은 조건을 검색
<=	작거나 같은 조건을 검색
BETWEEN a AND b	A 와 B사이에 있는 범위 값을 모두 검색
IN(a,b,c)	A 이거나 B 이거나 C 인 조건을 검색
Like	특정 패턴을 가지고 있는 조건을 검색
Is Null / Is Not Null	Null 값을 검색 / Null 이 아닌 값을 검색
A AND B	A 조건과 B 조건을 모두 만족하는 값만 검색
A OR B	A 조건이나 B 조건 중 한가지라도 만족하는 값을 검색
NOT A	A 가 아닌 모든 조건을 검색

SELECT

- ① 비교 연산자를 사용하여 Student 테이블에서 키가(height) 180 cm 보다 크거나 같은 사람을 출력하세요



```
oracle@localhost:~  
SCOTT>SELECT name , height  
2      FROM student  
3      WHERE height >= 180 ;
```

NAME	HEIGHT
서진수	180
박동호	182
일지매	182
노정호	184

```
SCOTT>
```


SELECT

- ② Between 연산자를 사용하여 Student 테이블에서 몸무게가 (weight) 60kg ~ 80kg 인 사람의 이름과 체중을 출력하세요.

```
oracle@localhost:~  
SCOTT>SELECT name , weight  
2 FROM student  
3 WHERE weight BETWEEN 60 AND 80 ;  
  
NAME                WEIGHT  
-----  
서진수              72  
서재수              64  
박동호              70  
일지매              72  
김진욱              70  
노정호              62  
안은수              63  
인영민              69  
  
8 rows selected.  
SCOTT>
```

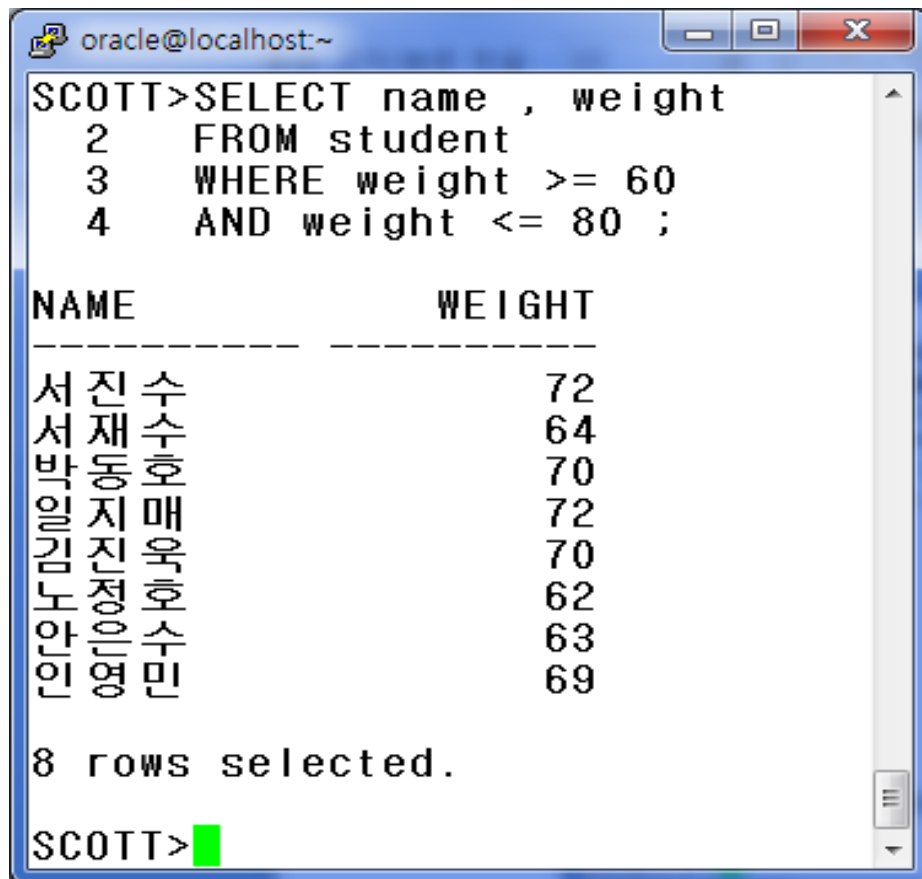
BETWEEN 연산자는 주의 사항이 있습니다.

1. 두 개의 값 중에 작은 값이 먼저 오고 큰 값이 나중에 와야 합니다.

2. 두 개의 값을 다 포함하여 출력됩니다

SELECT

- BETWEEN 연산자는 아래처럼 사용 가능합니다.



```
oracle@localhost:~  
SCOTT>SELECT name , weight  
2      FROM student  
3      WHERE weight >= 60  
4      AND weight <= 80 ;  
  
NAME                WEIGHT  
-----
```

NAME	WEIGHT
서진수	72
서재수	64
박동호	70
일지매	72
김진욱	70
노정호	62
안은수	63
인영민	69

```
8 rows selected.  
SCOTT>
```

SELECT

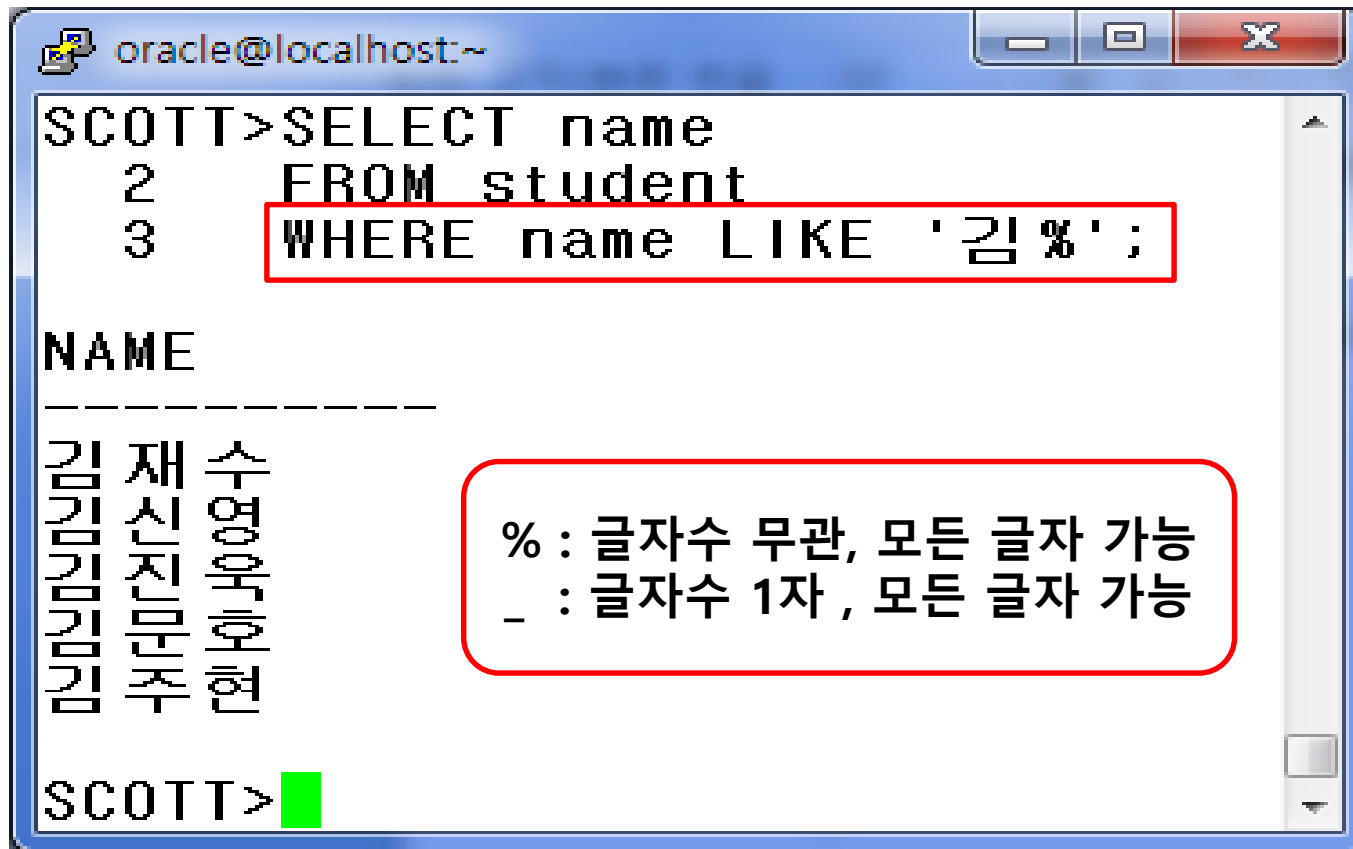
- ③ IN 연산자를 사용하여 Student 테이블에서 101 번 학과 학생과 201 번 학과 학생들을 모두 출력하세요

```
oracle@localhost:~  
SCOTT>SELECT name , deptno1  
2 FROM student  
3 WHERE deptno1 IN (101,201);  
  
NAME                DEPTNO1  
-----  
서진수              101  
김재수              201  
김신영              101  
임세현              201  
일지매              101  
안광훈              201  
김문호              201  
이윤나              101  
안은수              201  
인영민              201  
  
10 rows selected.  
  
SCOTT>
```

이 SQL 의 WHERE 절 부분을
" WHERE deptno1 = 101 OR
deptno1 = 201 " 로 사용할 수 도 있
지만 쿼리가 너무 길어져서 간편하게
IN 연산자를 사용하는 것입니다.

SELECT

- ④ Like 연산자를 사용하여 student 테이블에서 성이 “김” 씨인 사람을 조회하세요

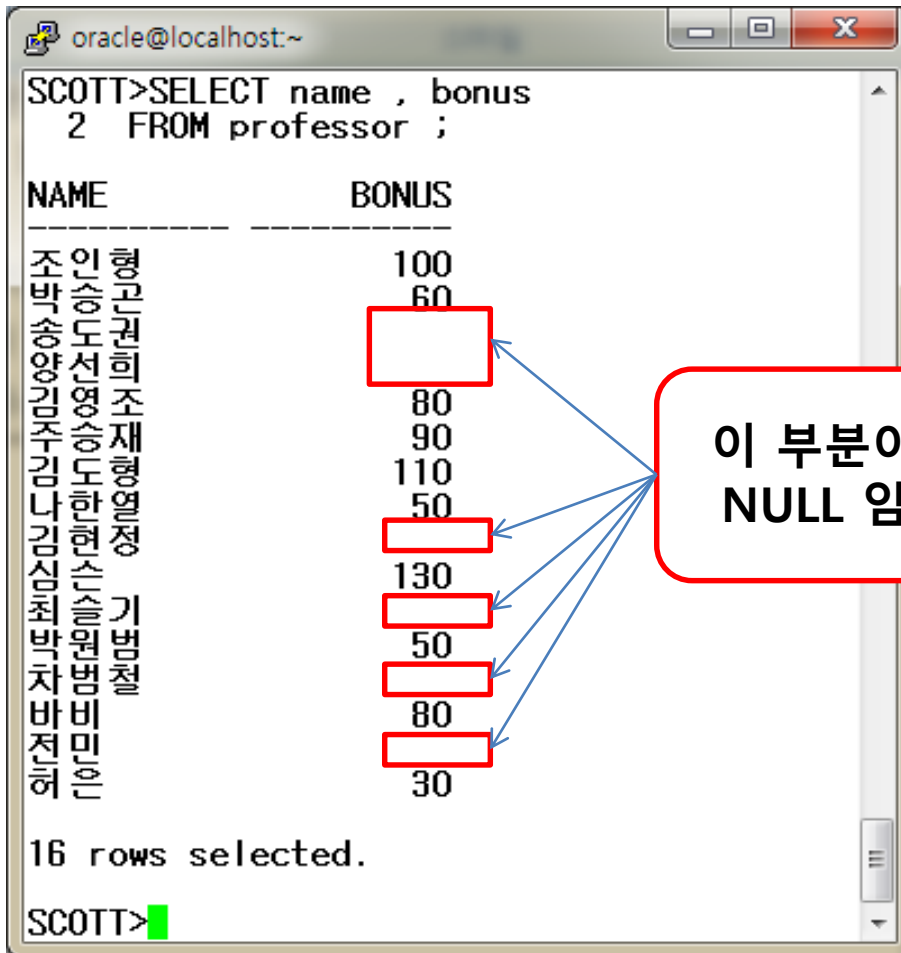


```
oracle@localhost:~  
SCOTT>SELECT name  
2      FROM student  
3      WHERE name LIKE '김%';  
  
NAME  
-----  
김재수  
김신영  
김진호  
김민영  
김주현  
  
SCOTT>
```

% : 글자수 무관, 모든 글자 가능
_ : 글자수 1자, 모든 글자 가능

SELECT

⑤ IS NULL / IS NOT NULL 연산자를 활용하기



oracle@localhost:~

```
SCOTT>SELECT name , bonus  
2 FROM professor ;
```

NAME	BONUS
조인성	100
박성민	60
송영준	80
양선영	90
김종도	110
김나한	50
심현수	130
최원빈	50
박기범	80
차비미	30
전지현	

16 rows selected.

SCOTT>

NULL 은 정해지지 않아서
값을 모른다는 의미임.
0 과는 다름 !!

이 부분이
NULL 임

SELECT

NULL 값만 출력하기

```
oracle@localhost:~  
SCOTT>SELECT name , bonus  
2 FROM professor  
3 WHERE bonus IS NULL ;  
  
NAME          BONUS  
-----  
송도권  
양선희  
김현정  
최승기  
차범철  
전민  
  
6 rows selected.  
SCOTT>
```

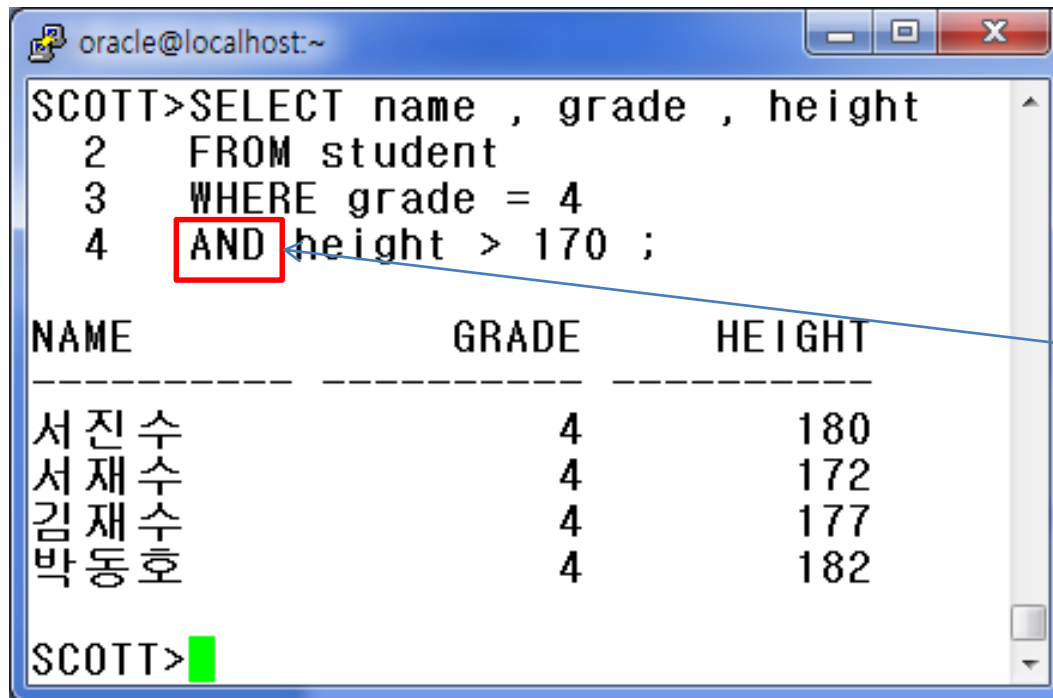
NULL 아닌 값만 출력하기

```
oracle@localhost:~  
SCOTT>SELECT name , bonus  
2 FROM professor  
3 WHERE bonus IS NOT NULL ;  
  
NAME          BONUS  
-----  
조인형  
박승진  
김영조  
주승재  
김도형  
나한영  
심스원  
박원범  
바비  
허은  
  
100  
60  
80  
90  
110  
50  
130  
50  
80  
30  
  
10 rows selected.  
SCOTT>
```

SELECT

⑥ 검색조건이 두 개 이상일 경우 조회하기

1. student 테이블을 사용하여 4학년 중에서 키가 170 cm 이상인 사람의 이름과 학년과 키를 조회하세요.



```
oracle@localhost:~  
SCOTT>SELECT name , grade , height  
2      FROM student  
3      WHERE grade = 4  
4      AND height > 170 ;  
  
NAME          GRADE    HEIGHT  
-----  
서진수        4         180  
서재수        4         172  
김재수        4         177  
박동호        4         182  
  
SCOTT>
```

The screenshot shows a terminal window with the Oracle SQL*Plus prompt. The user has entered a SELECT statement to retrieve names, grades, and heights from the 'student' table, filtered by grade 4 and height greater than 170. The 'AND' keyword in the WHERE clause is highlighted with a red box. The results are displayed in a table format with columns NAME, GRADE, and HEIGHT. The results show four students: 서진수 (180cm), 서재수 (172cm), 김재수 (177cm), and 박동호 (182cm).

두 가지 조건을 모두 만족하는 결과를 검색하실 때는 **AND 연산자**를 사용하여 조건을 적으시면 됩니다.

SELECT

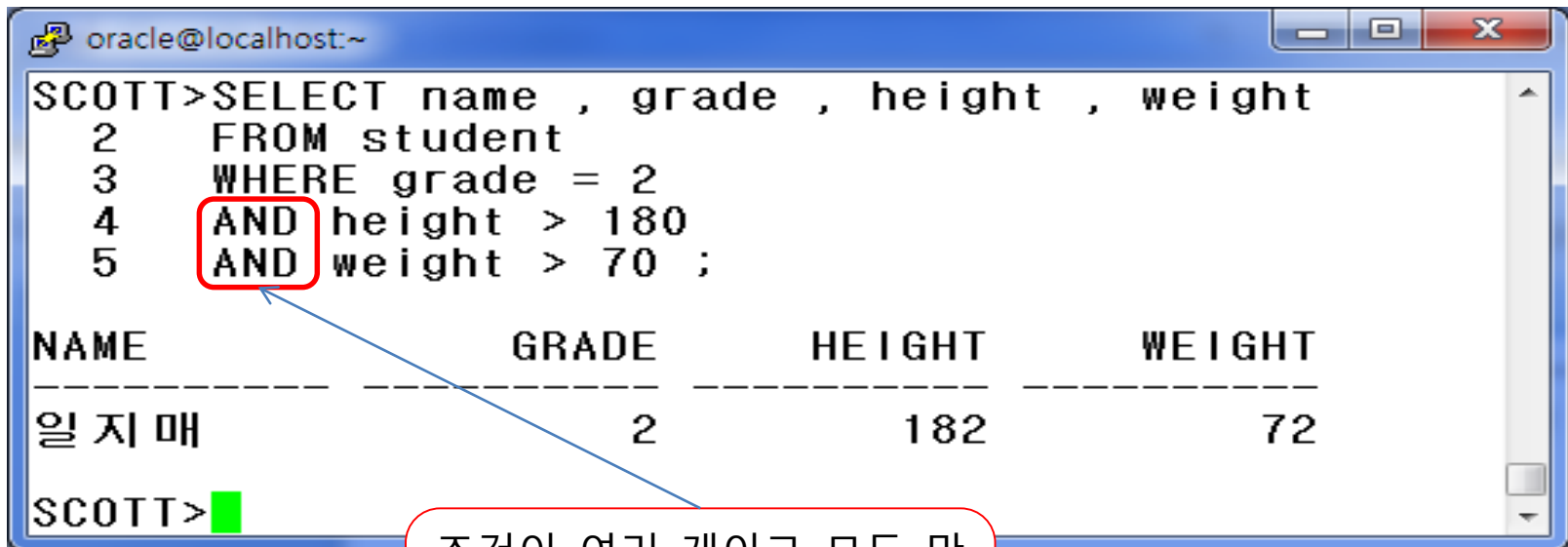
2. student 테이블을 사용하여 1학년 이거나 또는 몸무게가 80 kg 이상인 학생들의 이름과 키와 학년과 몸무게를 출력하세요.

```
oracle@localhost:~  
SCOTT>SELECT name , grade , height , weight  
2 FROM student  
3 WHERE grade = 1  
4 OR weight > 80 ;  
  
NAME                GRADE    HEIGHT    WEIGHT  
-----  
김재수              4        177       83  
안광훈              2        175       82  
이윤나              1        162       48  
안인수              1        175       63  
인영민              1        173       69  
김주현              1        179       81  
허우                1        163       51  
  
7 rows selected.  
  
SCOTT>
```

두 가지 조건
중 한 가지 만
만족하는 행을
검색하고 싶으
면 **OR** 연산자
를 사용하면
됩니다.

SELECT

3. Student 테이블을 사용하여 2학년 중에서 키가 180 cm 보다 크면서 몸무게가 70 kg 보다 큰 학생들의 이름과 학년과 키와 몸무게를 출력하세요.



The screenshot shows a terminal window titled 'oracle@localhost:~'. The prompt is 'SCOTT>'. The user has entered the following SQL query:

```
SCOTT>SELECT name , grade , height , weight
2      FROM student
3      WHERE grade = 2
4      AND height > 180
5      AND weight > 70 ;
```

The query results are displayed in a table format:

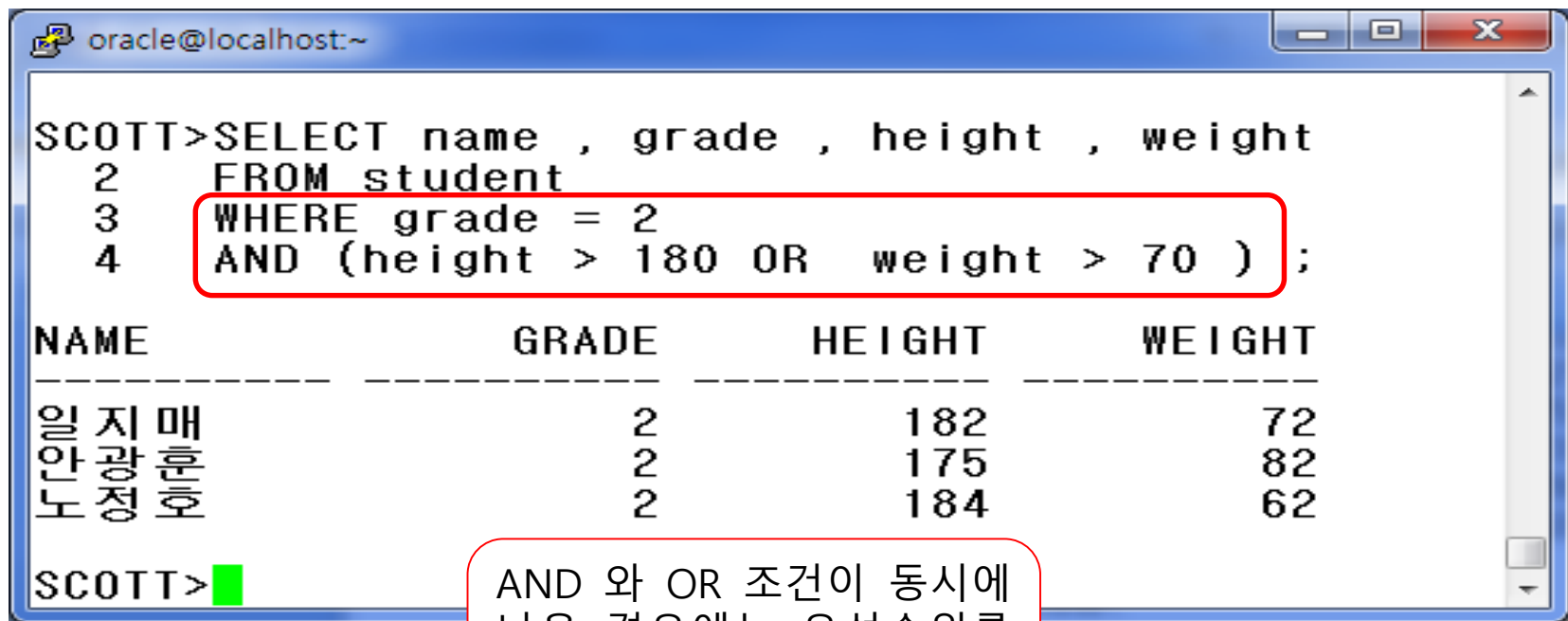
NAME	GRADE	HEIGHT	WEIGHT
일지매	2	182	72

The prompt 'SCOTT>' is followed by a green cursor. A red box highlights the 'AND' keyword in the query, and a blue arrow points from this box to a text box below the table.

조건이 여러 개이고 모두 만족하는 하는 경우 AND 를 여러 번 쓰면 됩니다

SELECT

4. Student 테이블을 사용하여 2학년 학생 중에서 키가 180 cm 보다 크거나 또는 몸무게가 70 kg 보다 큰 학생들의 이름과 학년과 키와 몸무게를 출력하세요.

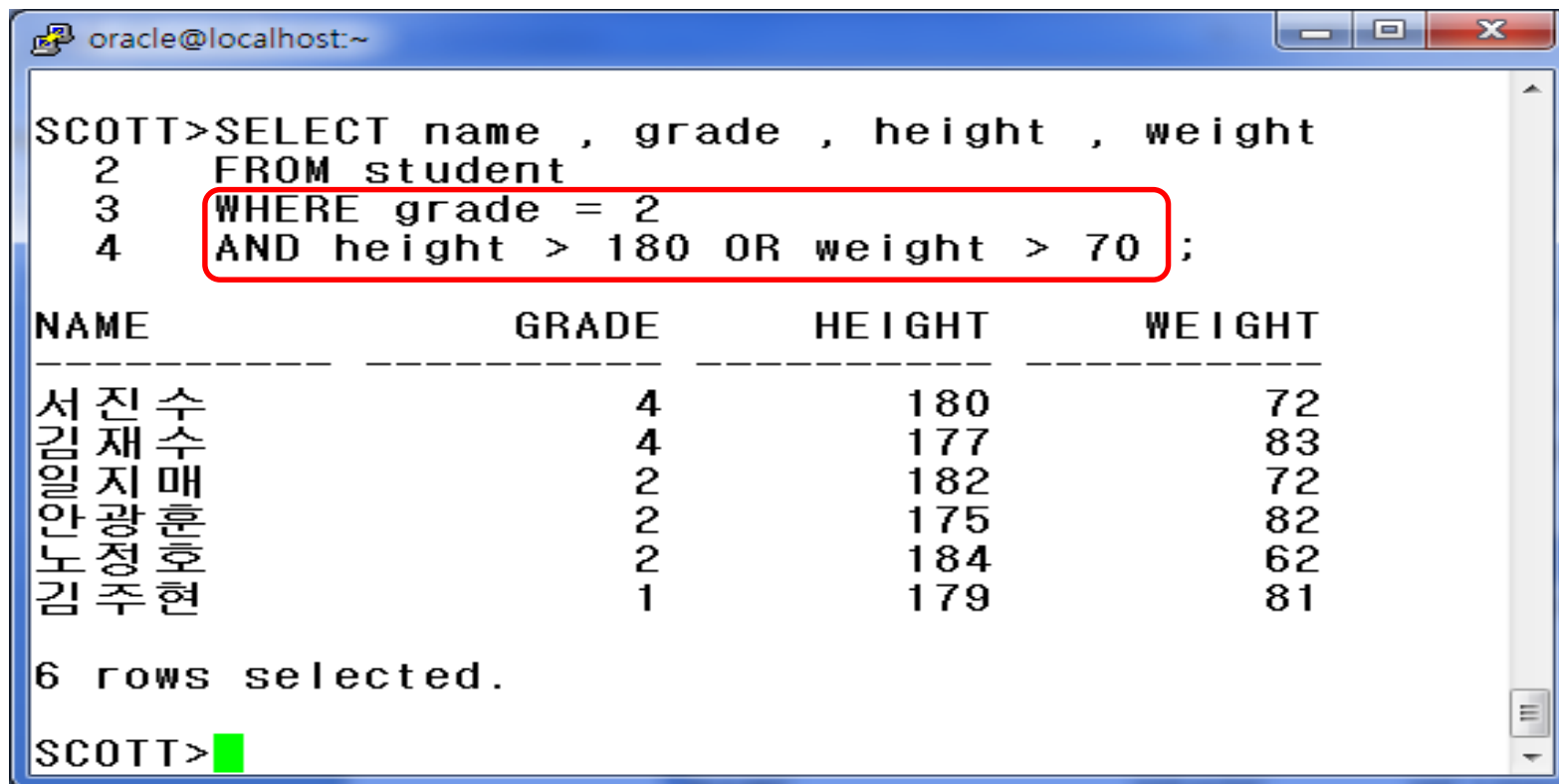


```
oracle@localhost:~  
SCOTT>SELECT name , grade , height , weight  
2      FROM student  
3      WHERE grade = 2  
4      AND (height > 180 OR  weight > 70 ) ;  
  
NAME                GRADE    HEIGHT    WEIGHT  
-----  
일지매                2        182        72  
안광훈                2        175        82  
노정호                2        184        62  
  
SCOTT>
```

AND 와 OR 조건이 동시에
나올 경우에는 우선순위를
아주 조심하셔야 합니다 !!!

SELECT

- AND 와 OR 연산자의 우선 순위 조절 실패 사례



The screenshot shows a terminal window titled 'oracle@localhost:~'. The user 'SCOTT' has entered the following SQL query:

```
SCOTT>SELECT name , grade , height , weight  
2 FROM student  
3 WHERE grade = 2  
4 AND height > 180 OR weight > 70 ;
```

The query results are displayed in a table with columns NAME, GRADE, HEIGHT, and WEIGHT. The results are as follows:

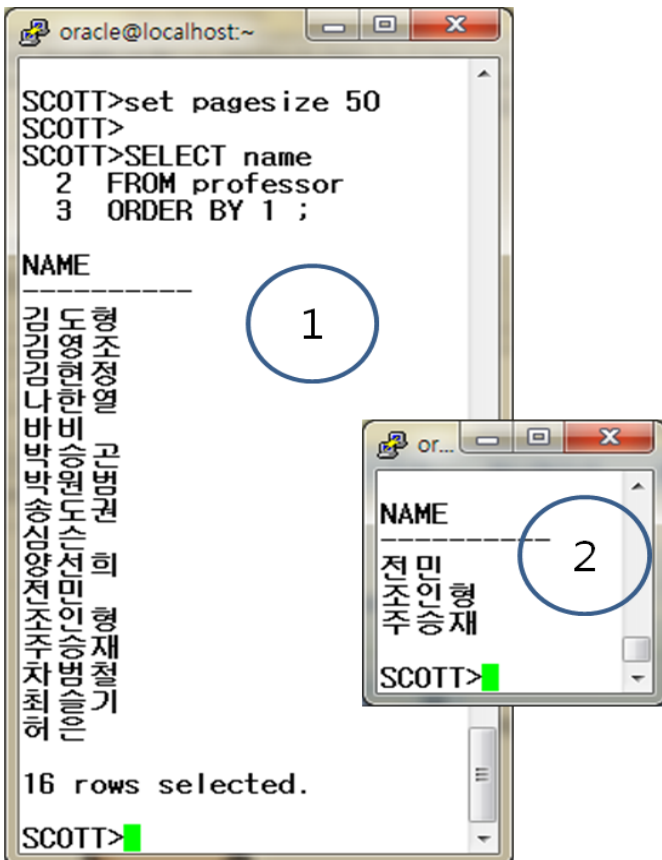
NAME	GRADE	HEIGHT	WEIGHT
서진수	4	180	72
김재수	4	177	83
일지매	2	182	72
안광훈	2	175	82
노정호	2	184	62
김주현	1	179	81

6 rows selected.
SCOTT>

SELECT

** 퀴즈 1 **

Professor 테이블에서 교수들의 이름을 조회하여 성 부분에 'ㅈ' 이 포함된 사람의 명단을 아래와 같이 출력하세요.



1번 화면이 professor 테이블에서 이름을 조회한 화면입니다. ORDER BY 라는 구문은 정렬을 해서 보여달라는 뜻인데 뒤에 살펴봅시다.

2번 화면이 성 부분에 'ㅈ' 이 들어간 사람만 출력한 화면입니다.

IN ('전민','조인형','주승재')

이렇게 하진 않으실 거죠?? 이런 거였으면 퀴즈도 안 냈을 겁니다~^^

능력을 보여주세요~~

SELECT

- Where name between '자%' and '차%'

SELECT

(10) ORDER BY 절을 사용하여 출력 결과 정렬하기

- 한 글: 가 , 나 , 다 , 라
- 영 어: A , B , C , D.....
- 숫 자: 1 , 2 , 3 , 4.....
- 날 짜: 예전 날짜부터 시작해서 최근 날짜로 정렬됨.
- ORDER BY 절을 사용하며 ASC 는 오름차순 , DESC 는 내림차순임.
- ASC 방식이 기본 정렬 방식임.

SELECT

- ① student 테이블을 사용하여 1 학년 학생의 이름과 키를 출력하세요.
단 키가 작은 순서대로 출력하세요.

```
oracle@localhost:~  
SCOTT>  
SCOTT>SELECT name , height  
2      FROM student  
3      WHERE grade = 1 ;
```

NAME	HEIGHT
이안나	162
안영민	175
김주현	173
허우	179
허우	163

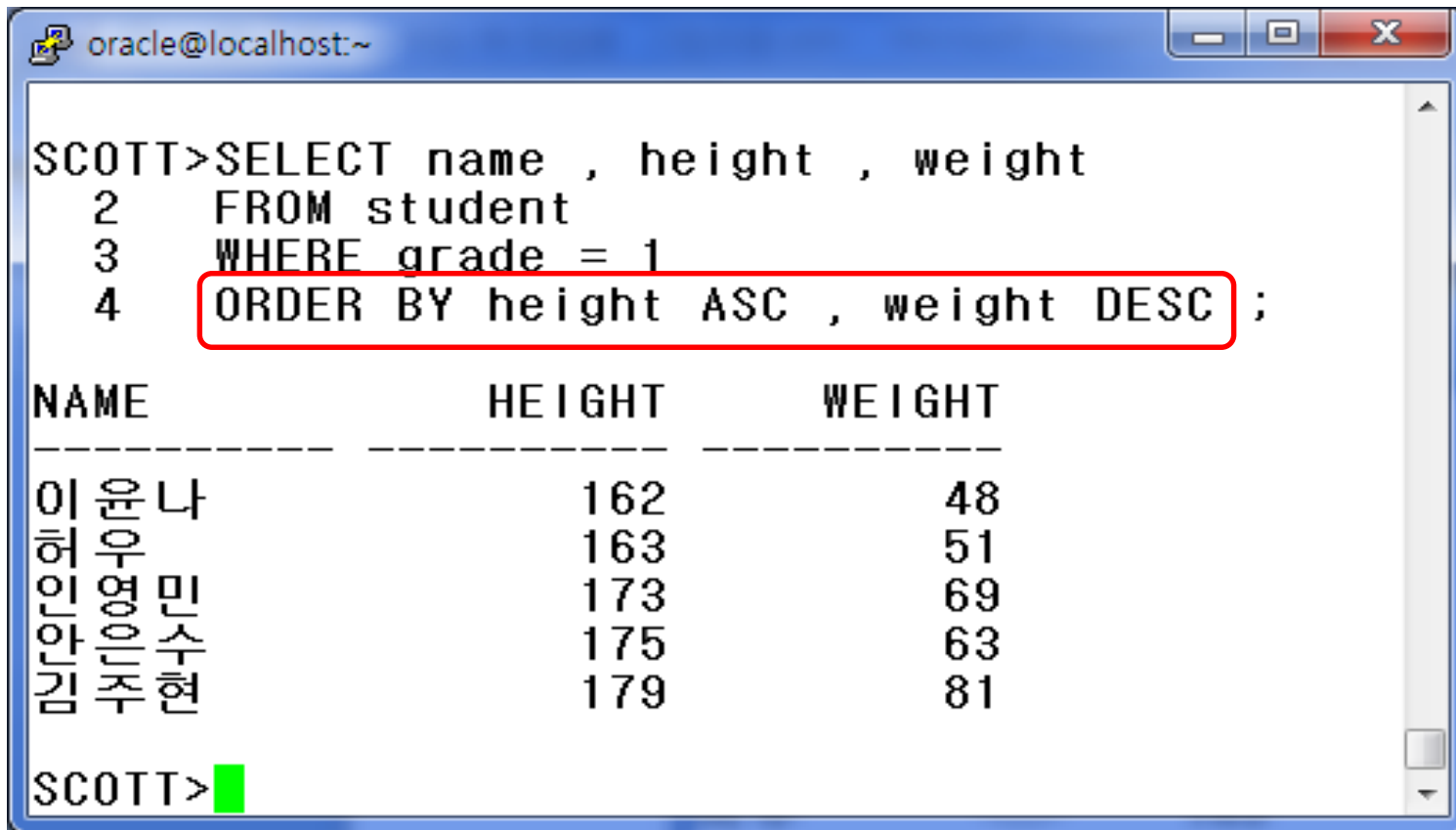
```
oracle@localhost:~  
SCOTT>SELECT name , height  
2      FROM student  
3      WHERE grade = 1  
4      ORDER BY height ;
```

NAME	HEIGHT
이안나	162
허우	163
안영민	173
김주현	175
허우	179

```
SCOTT>
```

SELECT

- ② Student 테이블을 사용하여 1학년 학생의 이름과 키와 몸무게를 출력하세요.
단 키는 작은 사람부터 출력하시고 몸무게는 많은 사람부터 출력하세요.



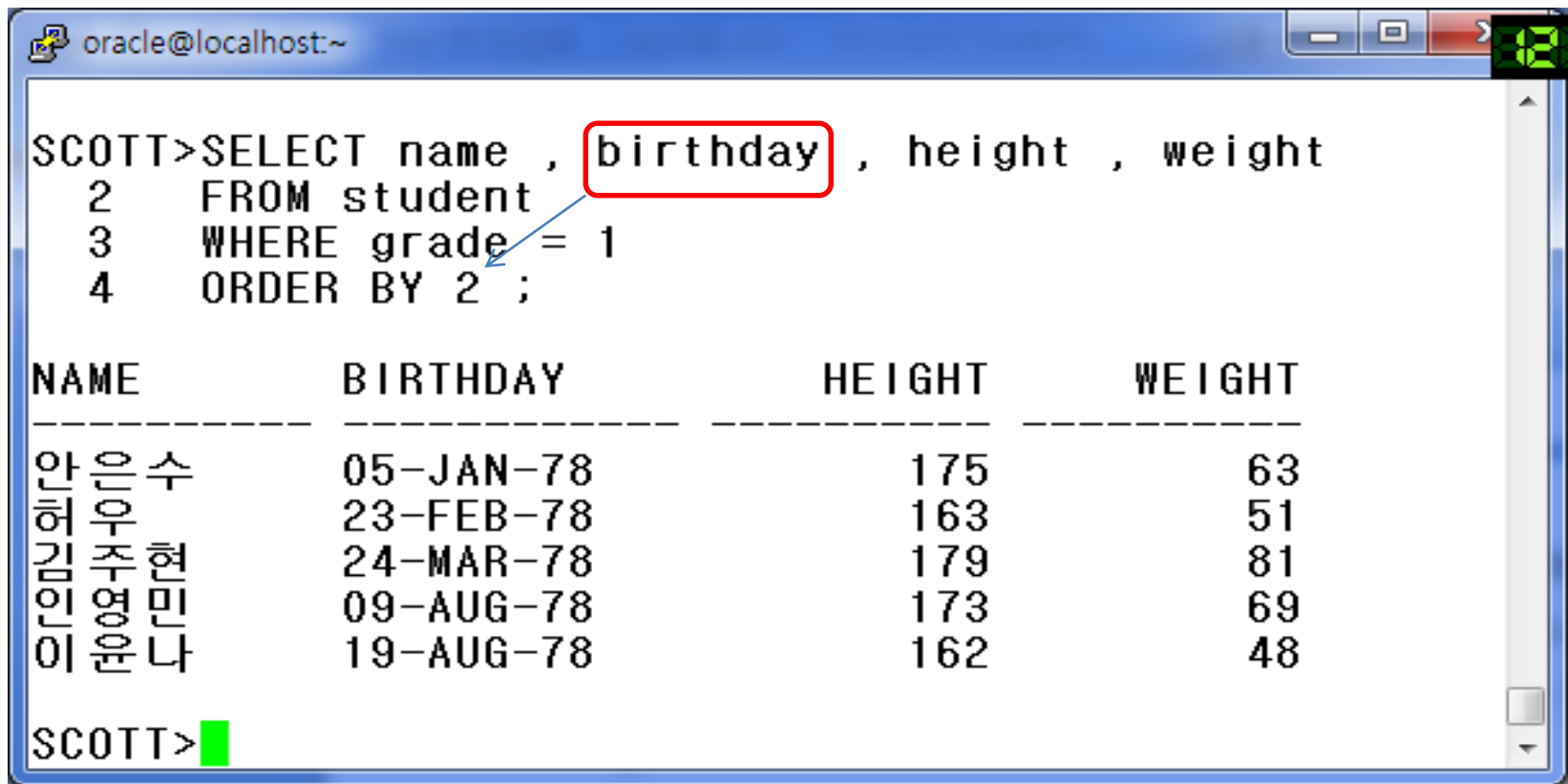
```
oracle@localhost:~  
SCOTT>SELECT name , height , weight  
2      FROM student  
3      WHERE grade = 1  
4      ORDER BY height ASC , weight DESC ;
```

NAME	HEIGHT	WEIGHT
이윤나	162	48
허아우	163	51
인영민	173	69
안이수	175	63
김주현	179	81

```
SCOTT>
```


SELECT

- ③ Student 테이블을 사용하여 1학년 학생의 이름과 생일과 키와 몸무게를 출력하세요. 단 생일이 빠른 사람 순서대로 정렬하세요.



The screenshot shows a terminal window titled 'oracle@localhost:~'. The user 'SCOTT' has entered the following SQL query:

```
SCOTT>SELECT name , birthday , height , weight
2   FROM student
3   WHERE grade = 1
4   ORDER BY 2 ;
```

The word 'birthday' in the query is highlighted with a red box. A blue arrow points from the '2' in 'ORDER BY 2' to the 'birthday' column in the result table below.

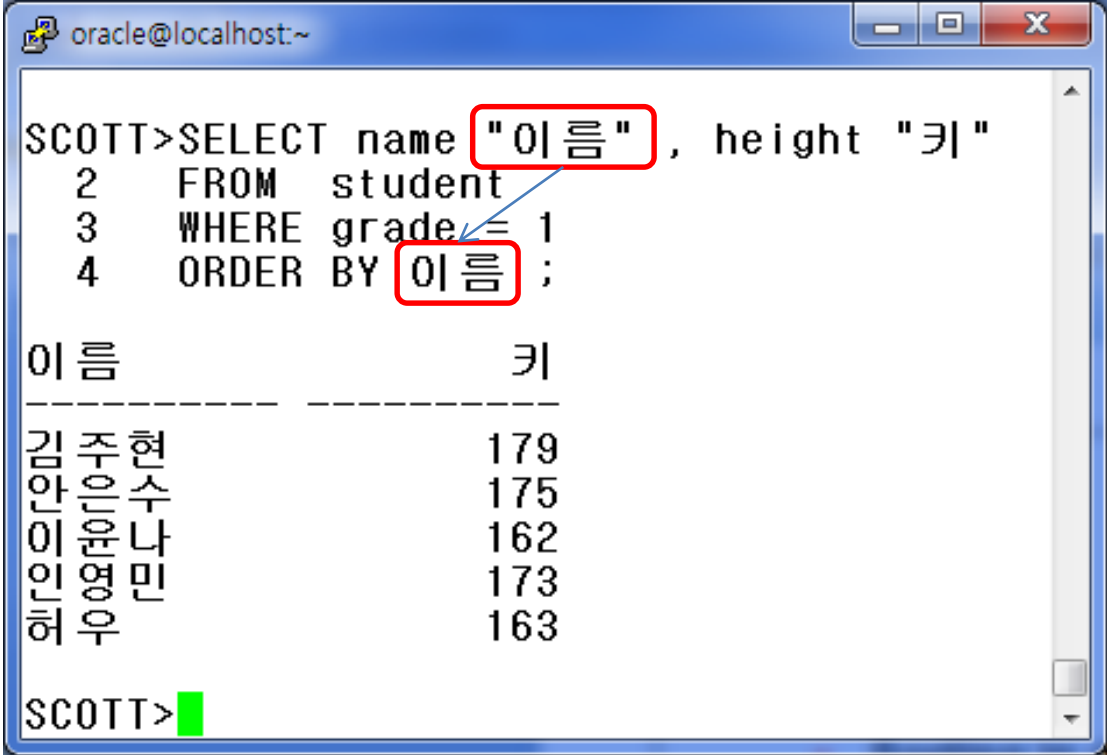
NAME	BIRTHDAY	HEIGHT	WEIGHT
안은수	05-JAN-78	175	63
허우	23-FEB-78	163	51
김주현	24-MAR-78	179	81
인영민	09-AUG-78	173	69
이윤나	19-AUG-78	162	48

The prompt 'SCOTT>' is followed by a green cursor bar.

SELECT

④ 칼럼의 별명을 사용한 정렬

Student 테이블을 사용하여 1학년 학생의 이름과 키를 출력하세요. 단 이름을 오름차순으로 정렬하세요.



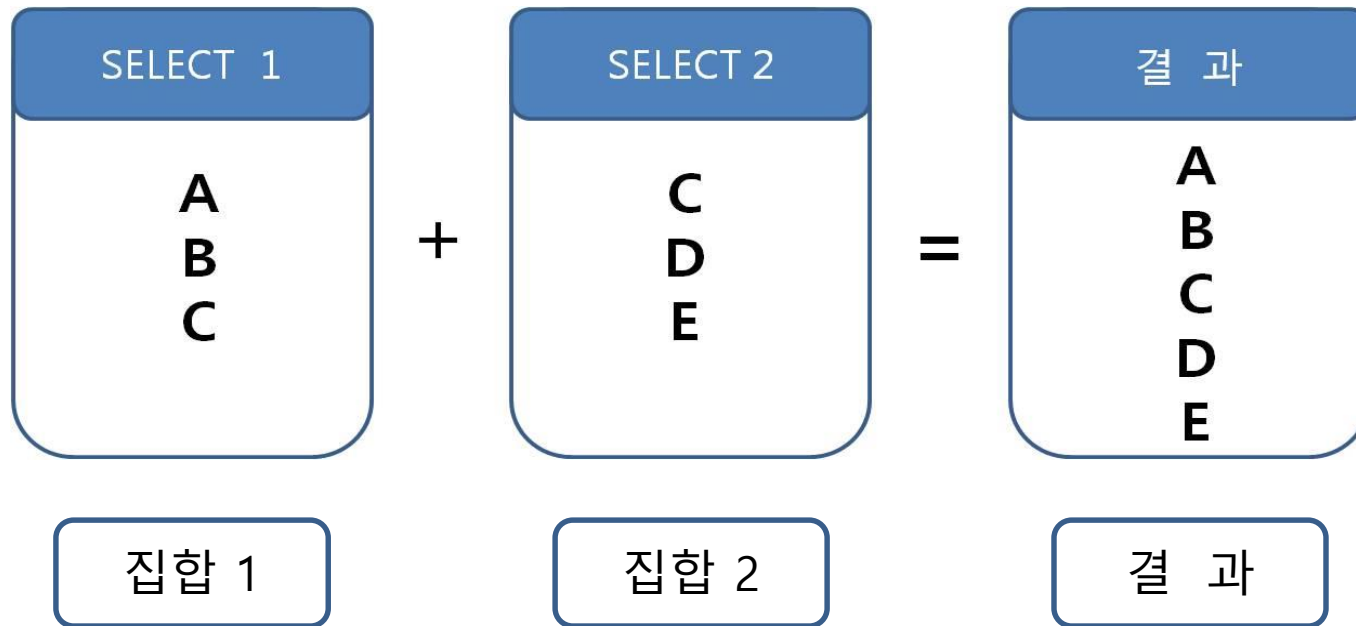
```
oracle@localhost:~  
SCOTT>SELECT name "이름", height "키"  
2      FROM student  
3      WHERE grade = 1  
4      ORDER BY 이름 ;  
  
이름      키  
-----  
김주현    179  
안수민    175  
이원나    162  
인영민    173  
허우      163  
  
SCOTT>
```

The screenshot shows a terminal window with the Oracle SQL*Plus prompt. The user has entered a SELECT query to retrieve the names and heights of first-grade students, ordered by name. The query uses aliases "이름" for name and "키" for height. The results are displayed in a table format with headers "이름" and "키". The data is as follows:

이름	키
김주현	179
안수민	175
이원나	162
인영민	173
허우	163

SELECT

(11) 집합 연산자 (Set Operator)



SELECT

- 집합 연산자 종류

연산자 종류	내 용
UNION	두 집합을 더해서 결과를 출력. 중복 값 제거하고 정렬함
UNION ALL	두 집합을 더해서 결과를 출력. 중복 값 제거 안하고 정렬 안 함
INTERSECT	두 집합의 교집합 결과를 출력함. 정렬함
MINUS	두 집합의 차집합 결과를 출력함. 정렬함. 쿼리의 순서 중요함

SELECT

① UNION / UNION ALL (두 집합을 더합니다)

-Student 테이블과 professor 테이블을 참조하여 101 번 학과에 소속되어 있는 학생과 교수들의 학번(교수님은 교수번호) , 이름, 학과번호를 출력하세요.

```
oracle@localhost:~  
SCOTT>SELECT studno , name , deptno1  
2 FROM student  
3 WHERE deptno1 = 101  
4 UNION  
5 SELECT profno , name, deptno  
6 FROM professor  
7 WHERE deptno = 101;  
  
STUDNO NAME DEPTN01  
-----  
1001 조인형 101  
1002 박승곤 101  
1003 송도권 101  
9411 서진수 101  
9511 김신영 101  
9611 일지매 101  
9711 이윤나 101  
  
7 rows selected.  
SCOTT>
```

```
oracle@localhost:~  
SCOTT>SELECT studno , name , deptno1  
2 FROM student  
3 WHERE deptno1 = 101  
4 UNION ALL  
5 SELECT profno , name, deptno  
6 FROM professor  
7 WHERE deptno = 101;  
  
STUDNO NAME DEPTN01  
-----  
9411 서진수 101  
9511 김신영 101  
9611 일지매 101  
9711 이윤나 101  
1001 조인형 101  
1002 박승곤 101  
1003 송도권 101  
  
7 rows selected.  
SCOTT>
```

SELECT

-Student 테이블에서 101번 학과와 201번 학과를 전공하는 학생들의 이름을 출력하세요.

```
oracle@localhost:~  
SCOTT>SELECT name  
2 FROM student  
3 WHERE deptno1=101  
4 UNION  
5 SELECT name  
6 FROM student  
7 WHERE deptno2=201 ;  
  
NAME  
-----  
김신영  
서진수  
김신영  
이윤나  
일지매  
SCOTT>
```

정렬함/중복
값 제거 후
출력

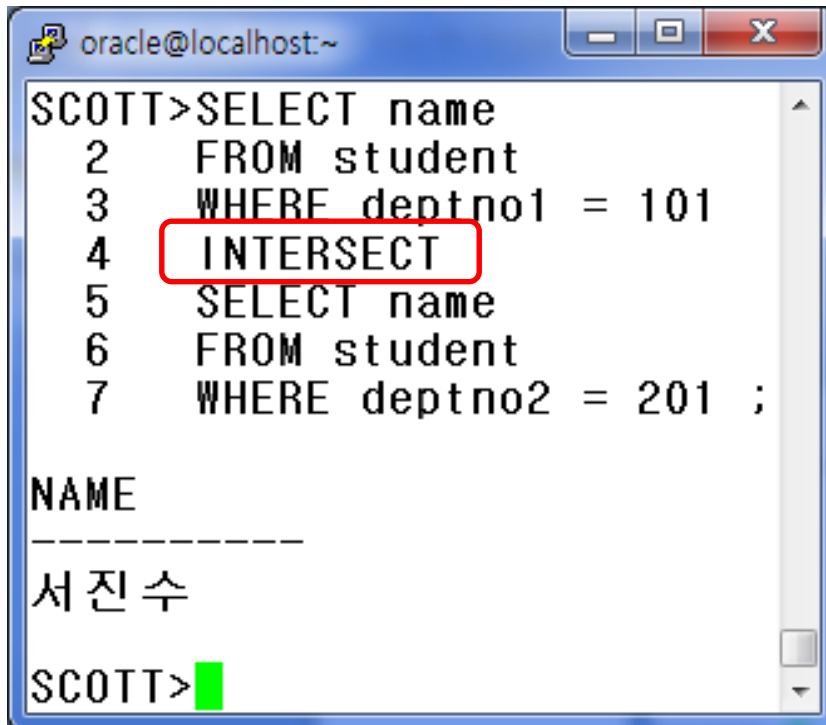
```
oracle@localhost:~  
SCOTT>SELECT name  
2 FROM student  
3 WHERE deptno1=101  
4 UNION ALL  
5 SELECT name  
6 FROM student  
7 WHERE deptno2=201 ;  
  
NAME  
-----  
서진수  
김신영  
일지매  
이윤나  
서진수  
김신영  
6 rows selected.  
SCOTT>
```

정렬 안 함/
중복값 모두
출력

SELECT

② INTERSECT 연산자 사용하기 – 교집합 찾기

-stduent 테이블을 사용하여 101 번 학과와 201 번 학과를 복수전공하는 사람의 이름을 출력하세요.



```
oracle@localhost:~  
SCOTT>SELECT name  
2     FROM student  
3     WHERE deptno1 = 101  
4     INTERSECT  
5     SELECT name  
6     FROM student  
7     WHERE deptno2 = 201 ;  
  
NAME  
-----  
서진수  
  
SCOTT>
```

SELECT

③ MINUS 연산자 사용하기 - 큰 집합에서 작은 집합 빼기

-professor 테이블에서 전체 직원의 급여를 20 % 인상하기 위한 직원 명단을 출력하려 합니다. 단 직급이 전임강사인 사람들은 명단에서 제외하세요.

[illegible]

The screenshot shows a SQL*Plus session with the following commands and output:

```

SCOTT>SELECT name , position
2 FROM professor
3 MINUS
4 SELECT name , position
5 FROM professor
6 WHERE position='전 임강사' ;

```

NAME	POSITION
김도형	교수
김영조	교수
김나한	교수
바비승	교수
박승원	교수
박승원	교수
심승원	교수
조승원	교수
최승원	교수
허승원	교수

11 rows selected.

SCOTT>

SELECT

**** 집합 연산자 사용시 주의사항**

1. 칼럼의 개수가 다를 경우 에러 발생함

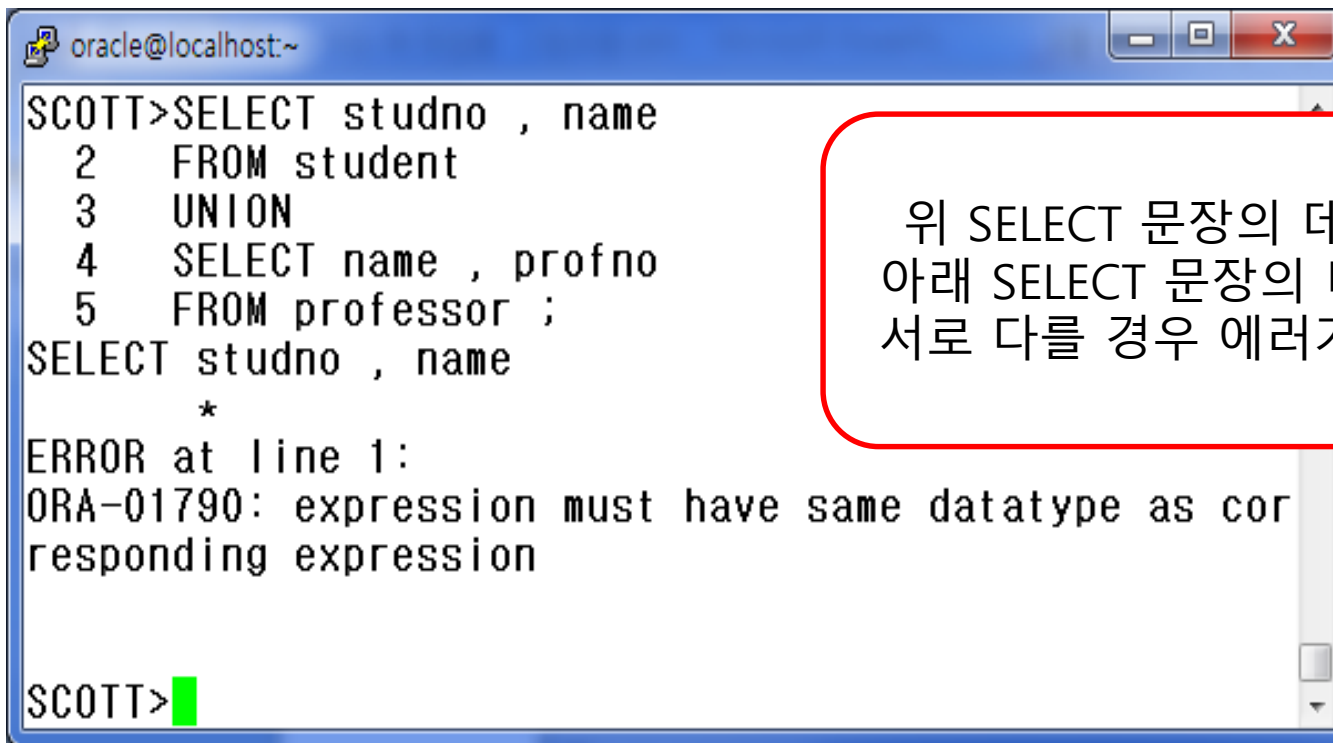
```
oracle@localhost:~  
SCOTT>SELECT studno , name  
2      FROM student  
3      UNION  
4      SELECT profno  
5      FROM professor;  
SELECT studno , name  
*  
ERROR at line 1:  
ORA-01789: query block has incorrect number  
of result columns  
  
SCOTT>
```

위 SELECT 절은 2개의 칼럼
인데 아래 SELECT 절은 1개
의 칼럼으로 칼럼의 개수가
다를 경우 에러가 발생합니다.

SELECT

** 집합 연산자 사용시 주의사항

2. 비교되는 칼럼끼리의 데이터 타입이 다를 경우



```
oracle@localhost:~  
SCOTT>SELECT studno , name  
2      FROM student  
3      UNION  
4      SELECT name , profno  
5      FROM professor ;  
SELECT studno , name  
      *  
ERROR at line 1:  
ORA-01790: expression must have same datatype as cor  
responding expression  
  
SCOTT>
```

위 SELECT 문장의 데이터 타입과
아래 SELECT 문장의 데이터 타입이
서로 다를 경우 에러가 발생합니다.