



08장. 서브 쿼리

이번 장에서는 역시 하나의 테이블에서 검색한 결과를 다른 테이블에 전달하여 새로운 결과를 검색하는 경우 사용하는 서브 쿼리로 다양한 검색을 해보도록 하겠습니다.

이 장에서 다룰 내용

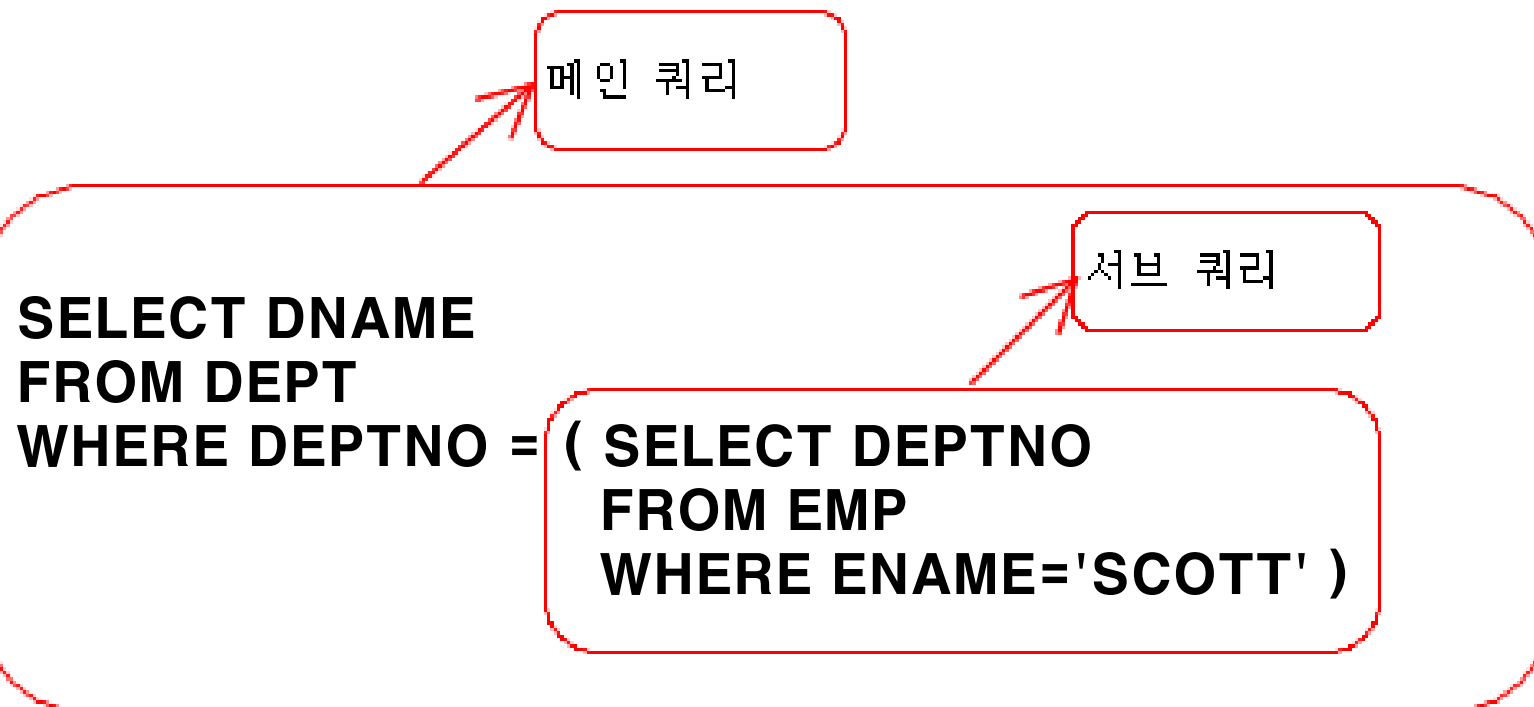


- 1 **서브 쿼리의 기본 개념**
- 2 **단일 행 서브 쿼리**
- 3 **서브 쿼리에서 그룹 함수의 사용**
- 4 **다중 행 서브 쿼리**

01. 서브 쿼리의 기본 개념



- ❖ SCOTT의 부서명을 알아내기 위한 서브 쿼리문 부터 살펴봅시다.



01. 서브 쿼리의 기본 개념



- ❖ 서브 쿼리는 하나의 SELECT 문장의 절 안에 포함된 또 하나의 SELECT 문장입니다.
- ❖ 그렇기에 서브 쿼리를 포함하고 있는 쿼리문을 메인 쿼리, 포함된 또 하나의 쿼리를 서브 쿼리라 합니다.
- ❖ 서브 쿼리는 비교 연산자의 오른쪽에 기술해야 하고 반드시 괄호로 둘러싸아야 합니다.
- ❖ 서브 쿼리는 메인 쿼리가 실행되기 이전에 한번만 실행이 됩니다.

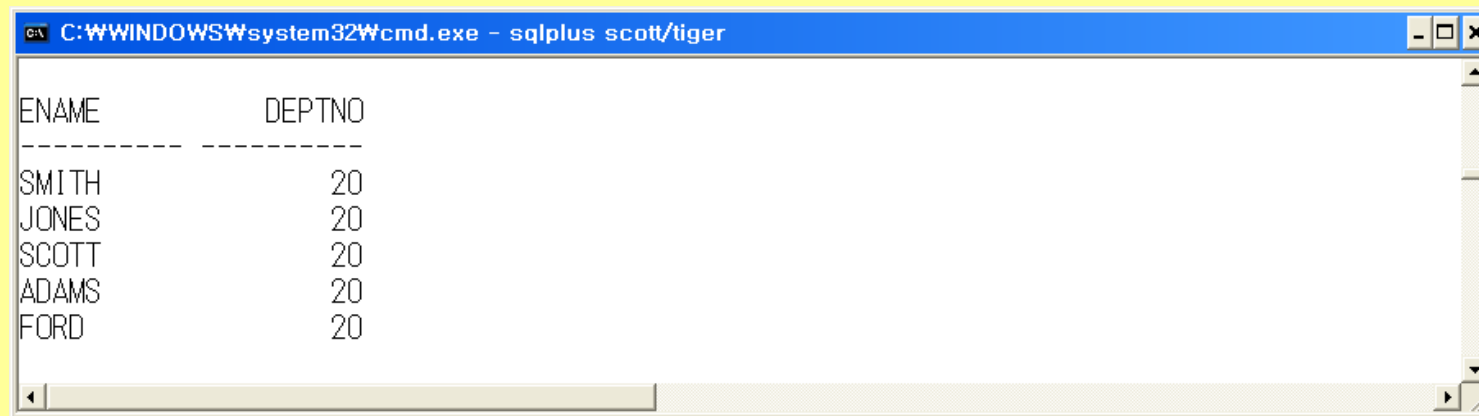
02. 단일 행 서브 쿼리



- ❖ 단일 행(Single Row) 서브 쿼리는 수행 결과가 오직 하나의 로우(행, row)만을 반환하는 서브 쿼리를 갖는 것을 말합니다.
- ❖ 단일 행 서브 쿼리문에서는 이렇게 오직 하나의 로우(행, row)로 반환되는 서브 쿼리의 결과는 메인 쿼리에 보내게 되는데 메인 쿼리의 WHERE 절에서는 단일 행 비교 연산자인 =, >, >=, <, <=, <>를 사용해야 합니다.

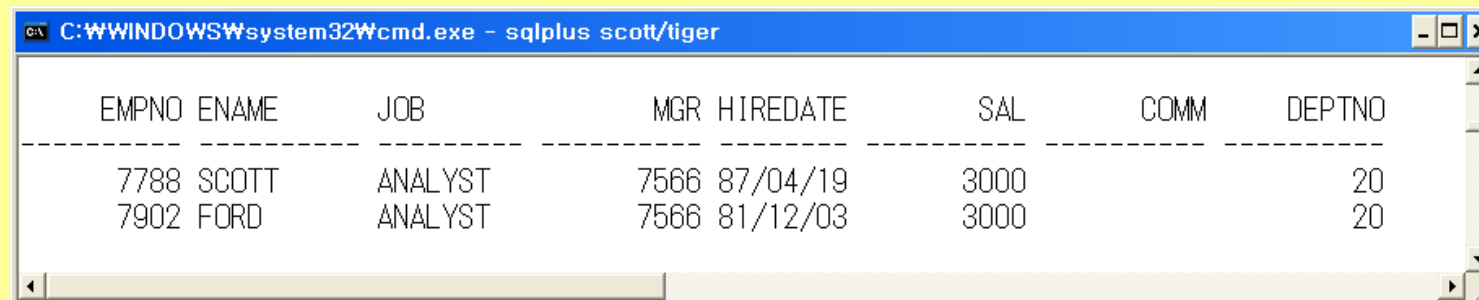
<탄탄히 다지기>

1. SCOTT과 같은 부서에서 근무하는 사원의 이름과 부서 번호를 출력하는 SQL 문을 작성해 보시오.



ENAME	DEPTNO
SMITH	20
JONES	20
SCOTT	20
ADAMS	20
FORD	20

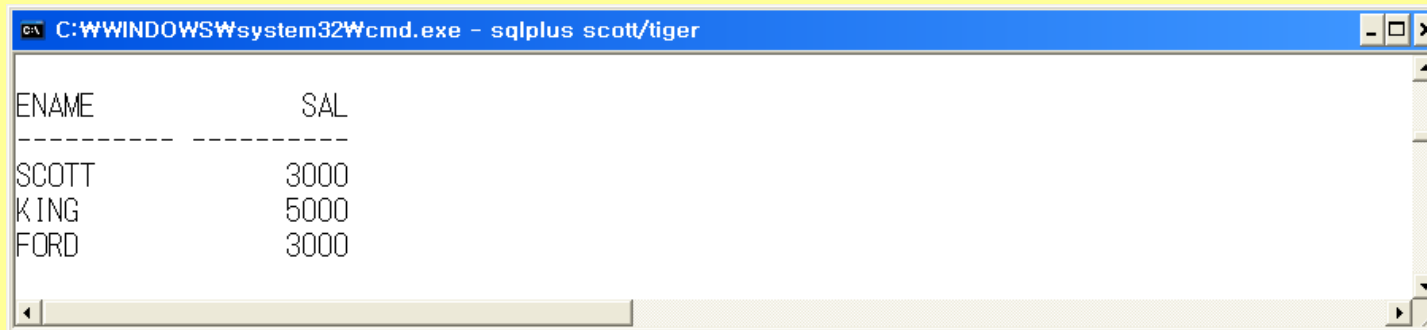
2. SCOTT와 동일한 직급을 가진 사원을 출력하는 SQL 문을 작성해 보시오.



EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO
7788	SCOTT	ANALYST	7566	87/04/19	3000		20
7902	FORD	ANALYST	7566	81/12/03	3000		20

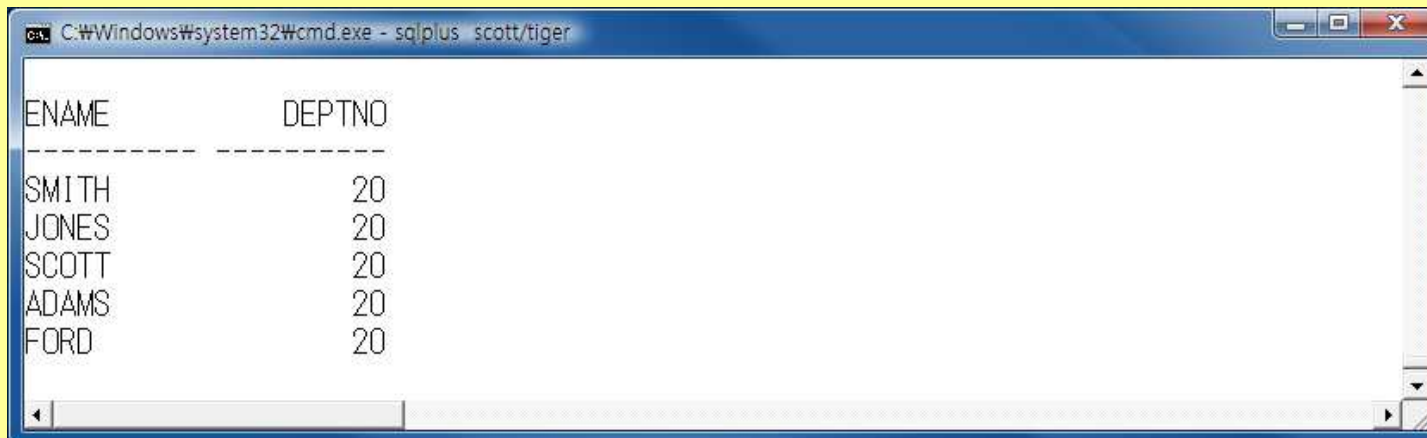
<탄탄히 다지기>

3. SCOTT의 급여와 동일하거나 더 많이 받는 사원명과 급여를 출력하시오.



ENAME	SAL
SCOTT	3000
KING	5000
FORD	3000

4. DALLAS에서 근무하는 사원의 이름, 부서 번호를 출력하시오.



ENAME	DEPTNO
SMITH	20
JONES	20
SCOTT	20
ADAMS	20
FORD	20

<탄탄히 다지기>

5. SALES(영업부) 부서에서 근무하는 모든 사원의 이름과 급여를 출력하시오.

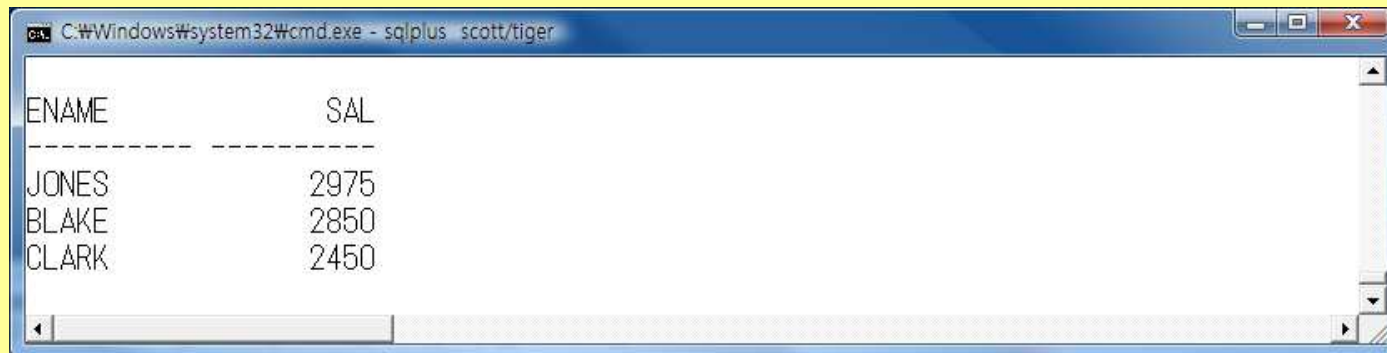


The screenshot shows a SQL*Plus window with the title bar "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The query results are displayed in a table format with columns ENAME and SAL. The data includes six rows for employees in the SALES department. At the bottom of the window, a status bar indicates "6 개의 행이 선택되었습니다." (6 rows selected).

ENAME	SAL
ALLEN	1600
WARD	1250
MARTIN	1250
BLAKE	2850
TURNER	1500
JAMES	950

6 개의 행이 선택되었습니다.

6. 직속상관이 KING인 사원의 이름과 급여를 출력하시오.



The screenshot shows a SQL*Plus window with the title bar "C:\Windows\system32\cmd.exe - sqlplus scott/tiger". The query results are displayed in a table format with columns ENAME and SAL. The data includes three rows for employees whose direct supervisor is KING.

ENAME	SAL
JONES	2975
BLAKE	2850
CLARK	2450

03. 서브 쿼리에서 그룹 함수의 사용

- ❖ 평균 급여를 구하는 쿼리문을 서브 쿼리로 사용하여 평균 급여보다 더 많은 급여를 받는 사원을 검색하는 문장은 다음과 같습니다.



```
SELECT ENAME, SAL
FROM EMP
WHERE SAL > ( SELECT AVG(SAL)
               FROM EMP );
```

```
C:\Windows\system32\cmd.exe - sqlplus scott/tiger

SQL> SELECT ENAME, SAL
2  FROM EMP
3  WHERE SAL > ( SELECT AVG(SAL)
4                FROM EMP );

ENAME          SAL
-----
JONES          2975
BLAKE          2850
CLARK          2450
SCOTT          3000
KING           5000
FORD           3000

6 개의 행이 선택되었습니다.

SQL>
```

04. 다중 행 서브 쿼리



- ❖ 다중 행 서브 쿼리는 서브 쿼리에서 반환되는 결과가 하나 이상의 행일 때 사용하는 서브 쿼리입니다. 다중 행 서브 쿼리는 반드시 다중 행 연산자(Multiple Row Operator)와 함께 사용해야 합니다.

종류	의미
IN	메인 쿼리의 비교 조건('=' 연산자로 비교할 경우)이 서브 쿼리의 결과 중에서 하나라도 일치하면 참입니다.
ANY, SOME	메인 쿼리의 비교 조건이 서브 쿼리의 검색 결과와 하나 이상이 일치하면 참입니다.
ALL	메인 쿼리의 비교 조건이 서브 쿼리의 검색 결과와 모든 값이 일치하면 참입니다.
EXIST	메인 쿼리의 비교 조건이 서브 쿼리의 결과 중에서 만족하는 값이 하나라도 존재하면 참입니다.

04. 다중 행 서브 쿼리

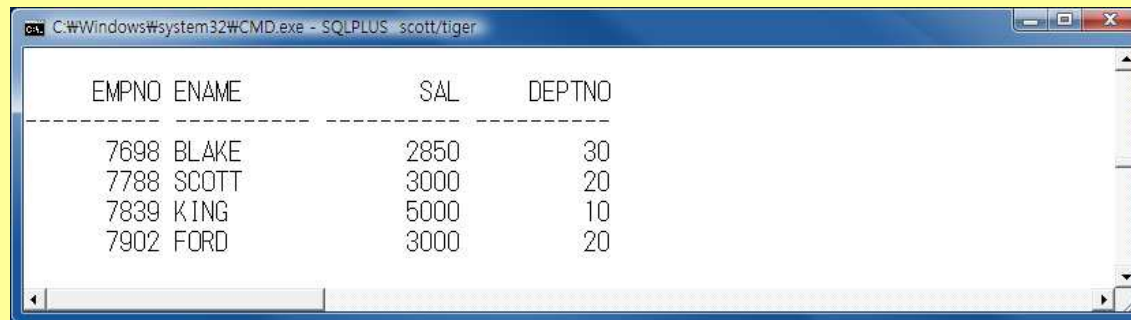


- ❖ 결과가 2개 이상 구해지는 쿼리문을 서브 쿼리로 기술할 경우에는 다중 행 연산자와 함께 사용해야 합니다.
- ❖ 주어진 문제가 3000 이상 받는 사원이 소속된 부서(10번, 20번)와 동일한 부서에서 근무하는 사원이기에 서브 쿼리의 결과 중에서 하나라도 일치하면 참인 결과를 구하는 IN 연산자와 함께 사용되어야 합니다.

```
SELECT ENAME, SAL, DEPTNO
FROM EMP
WHERE DEPTNO IN ( SELECT DISTINCT DEPTNO
FROM EMP
WHERE SAL>=3000);
```

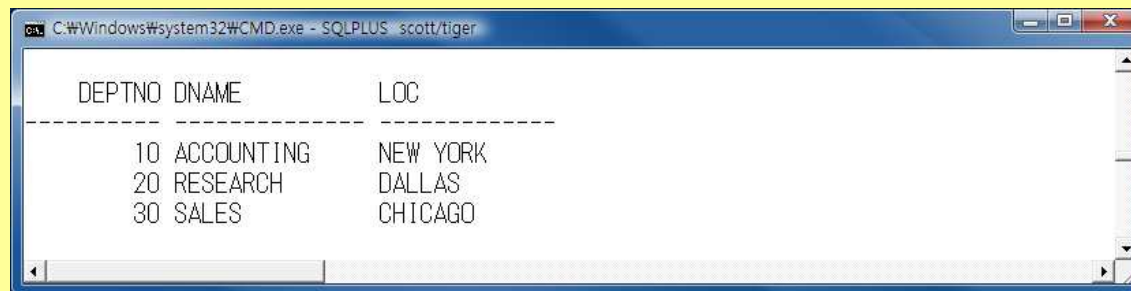
<탄탄히 다지기>

7. 부서별로 가장 급여를 많이 받는 사원의 정보(사원 번호, 사원 이름, 급여, 부서 번호)를 출력하시오.(IN 연산자 이용)



EMPNO	ENAME	SAL	DEPTNO
7698	BLAKE	2850	30
7788	SCOTT	3000	20
7839	KING	5000	10
7902	FORD	3000	20

8. 직급(JOB)이 **MANAGER**인 사람의 속한 부서의 부서 번호와 부서명과 지역을 출력하시오.



DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO

4.2 ALL 연산자



- ❖ ALL 조건은 메인 쿼리의 비교 조건이 서브 쿼리의 검색 결과와 모든 값이 일치하면 참입니다.
- ❖ 찾아진 값에 대해서 AND 연산을 해서 모두 참이면 참이 되는 셈이 됩니다. > ALL 은 “모든 비교값 보다 크냐” 고 묻는 것이 되므로 최대값보다 더 크면 참이 됩니다.

4.2 ALL 연산자

- ❖ 30번 소속 직원들 중에서 급여를 가장 많이 받는 직원보다 더 많은 급여를 받는 사람의 이름, 급여를 출력하는 쿼리문을 작성해 보시다.



```
SELECT ENAME, SAL
FROM EMP
WHERE SAL > ALL(SELECT SAL
                  FROM EMP
                  WHERE DEPTNO =30);
```

The screenshot shows a Windows command prompt window titled "C:\Windows\system32\CMD.exe - SQLPLUS scott/tiger". The prompt is "SQL>". The user has entered the following SQL query:

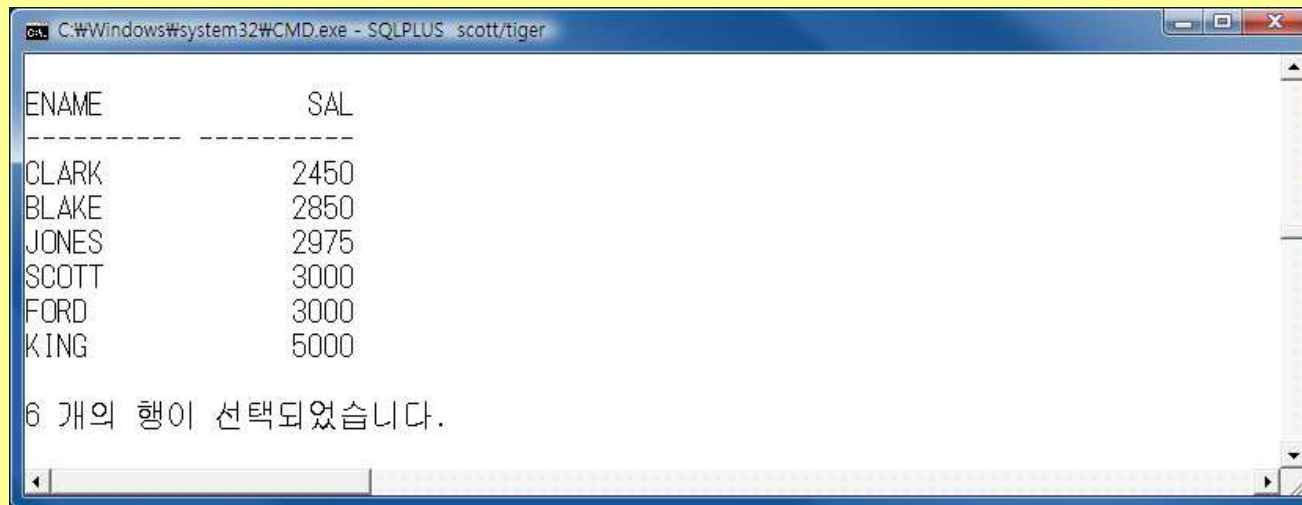
```
SQL> SELECT ENAME, SAL
2 FROM EMP
3 WHERE SAL > ALL(SELECT SAL
4                 FROM EMP
5                 WHERE DEPTNO =30);
```

The query results are displayed in a table format:

ENAME	SAL
JONES	2975
SCOTT	3000
FORD	3000
KING	5000

<탄탄히 다지기>

9. 영업 사원들 보다 급여를 많이 받는 사원들의 이름과 급여와 직급(담당 업무)를 출력하되 영업 사원은 출력하지 않습니다.



```
C:\Windows\system32\CMD.exe - SQLPLUS scott/tiger

ENAME          SAL
-----
CLARK          2450
BLAKE          2850
JONES          2975
SCOTT          3000
FORD           3000
KING           5000

6 개의 행이 선택되었습니다.
```

4.3 ANY 연산자



- ❖ ANY 조건은 메인 쿼리의 비교 조건이 서브 쿼리의 검색 결과와 하나 이상만 일치하면 참입니다.
- ❖ > ANY는 찾아진 값에 대해서 하나라도 크면 참이 되는 셈이 됩니다. 그러므로 찾아진 값 중에서 가장 작은 값 즉, 최소값 보다 크면 참이 됩니다.

4.3 ANY 연산자

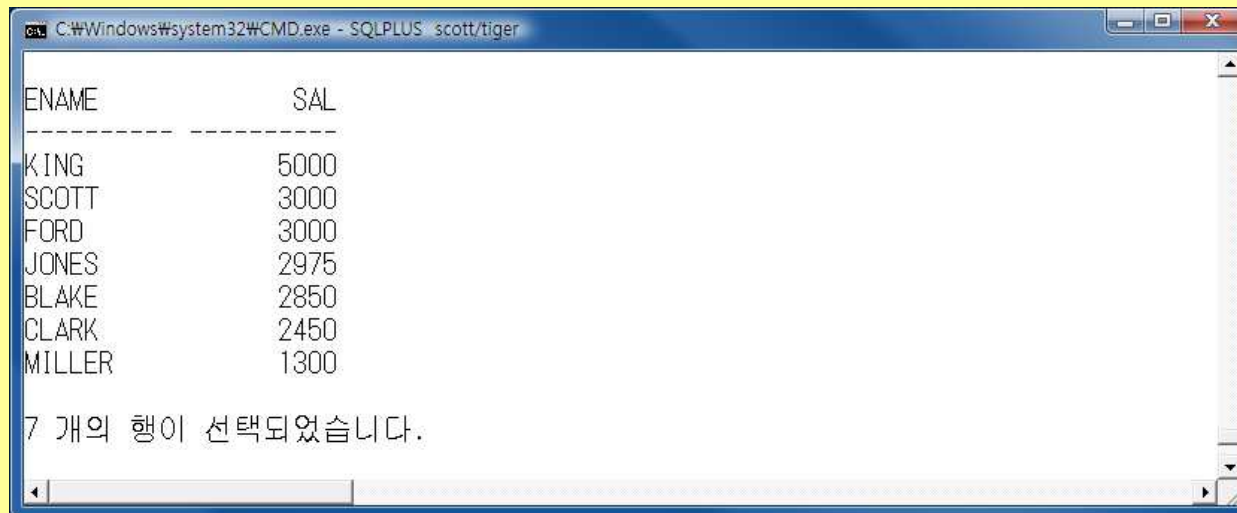


- ❖ 다음은 부서번호가 30번인 직원들의 급여 중 가장 작은 값(950)보다 많은 급여를 받는 직원의 이름, 급여를 출력하는 예제를 작성해 봅시다.

```
SELECT ENAME, SAL
FROM EMP
WHERE SAL > ANY ( SELECT SAL
                   FROM EMP
                   WHERE DEPTNO = 30 );
```

<탄탄히 다지기>

10. 영업 사원들의 최소 급여를 많이 받는 사원들의 이름과 급여와 직급(담당 업무)를 출력하되 영업 사원은 출력하지 않습니다.



The screenshot shows a Windows command prompt window titled "C:\Windows\system32\CMD.exe - SQLPLUS scott/tiger". The window displays the output of a SQL query. The output is a table with two columns: "ENAME" and "SAL". The data rows are: KING (5000), SCOTT (3000), FORD (3000), JONES (2975), BLAKE (2850), CLARK (2450), and MILLER (1300). Below the table, a message states "7 개의 행이 선택되었습니다." (7 rows selected).

ENAME	SAL
KING	5000
SCOTT	3000
FORD	3000
JONES	2975
BLAKE	2850
CLARK	2450
MILLER	1300

7 개의 행이 선택되었습니다.



Thank You !

Dynamic_오라클 11g + PL/SQL 입문 8장