



13강.oracle

- 조인을 사용

1

조인을 사용하여 다중 테이블에서 데이터 표시

목차

1 조인의 필요성



2 Cross Join



3 Equi Join



4 Non-Equi Join



1 조인을 사용하여 다중 테이블에서 데이터 표시

조인의 필요성

- ❖ 특정 부서 번호에 대한 부서이름은 무엇인지는 부서(DEPT) 테이블에 있습니다. 특정 사원에 대한 부서명을 알아내기 위해서는 부서 테이블에서 정보를 얻어 와야 합니다.

The image displays two SQL command windows side-by-side, illustrating a join operation. The left window shows a query of employees by department number, and the right window shows a query of department names by department number. Red arrows connect the department numbers in the first query to the department names in the second query.

Left Window (EMP Table):

```
SQL> SELECT ENAME, DEPTNO
2 FROM EMP
3 ORDER BY DEPTNO;
```

ENAME	DEPTNO
CLARK	10
KING	10
MILLER	10
JONES	20
FORD	20
ADAMS	20
SMITH	20
SCOTT	20
WARD	30
TURNER	30
ALLEN	30
JAMES	30
BLAKE	30
MARTIN	30

14 개의 행이 선택되었습니다.

Right Window (DEPT Table):

```
SQL> SELECT DEPTNO, DNAME
2 FROM DEPT;
```

DEPTNO	DNAME
10	ACCOUNTING
20	RESEARCH
30	SALES
40	OPERATIONS

1 조인을 사용하여 다중 테이블에서 데이터 표시

조인의 필요성

- ❖ SCOTT인 사원이 소속되어 있는 부서의 이름이 무엇인지 알아보려고 합니다.
- ❖ SCOTT이란 사원의 부서명을 알아내는 일 역시 사원 테이블에서 SCOTT이 소속된 부서 번호를 알아낸 후에 부서 테이블에서

해당

```
C:\Windows\system32\cmd.exe
SQL> SELECT DEPTNO
2 FROM EMP
3 WHERE ENAME='SCOTT';

DEPTNO
-----
      20

SQL>
```

서명

```
C:\Windows\system32\cmd.exe - sqlplus scot...
SQL> SELECT *
2 FROM DEPT
3 WHERE DEPTNO=20;

DEPTNO DNAME          LOC
-----
      20 RESEARCH      DALLAS

SQL>
```

- ❖ 실습에서처럼 원하는 정보가 두 개 이상의 테이블에 나누어져 있다면 위와 같이 여러 번 질의를 해야 할까요?

1

조인을 사용하여 다중 테이블에서 데이터 표시

Cross join

- ❖ 다음은 Cross Join으로 특별한 키워드 없이 SELECT 문의 FROM 절에 사원(EMP) 테이블과 부서(DEPT) 테이블을 콤마로 연결하여 연속하여 기술하는 것입니다.

예

```
SELECT *
FROM EMP, DEPT;
```

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

```
SQL> SELECT *
2 FROM EMP, DEPT;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DEPTNO DNAME	LOC
7369	SMITH	CLERK	7902	09/12/17	800		20	10 ACCOUNTING NEW YORK	
7489	ALLEN	SALESMAN	7698	01/02/20	1600	300	30	10 ACCOUNTING NEW YORK	
7521	WARD	SALESMAN	7698	01/02/22	1250	500	30	10 ACCOUNTING NEW YORK	
7566	JONES	MANAGER	7698	01/04/02	2975		20	10 ACCOUNTING NEW YORK	
7654	MARTIN	SALESMAN	7698	01/09/20	1250	1400	30	10 ACCOUNTING NEW YORK	
7698	BLAKE	MANAGER	7698	01/05/01	2850		30	10 ACCOUNTING NEW YORK	
7702	CLARK	MANAGER	7698	01/06/09	2450		10	10 ACCOUNTING NEW YORK	
7788	SCOTT	ANALYST	7566	07/04/19	3000		20	10 ACCOUNTING NEW YORK	
7839	KING	PRESIDENT		01/11/17	5000		10	10 ACCOUNTING NEW YORK	
7844	TURNER	SALESMAN	7698	01/09/00	1500	0	30	10 ACCOUNTING NEW YORK	
7876	ADAMS	CLERK	7788	07/05/23	1100		20	10 ACCOUNTING NEW YORK	
7900	JAMES	CLERK	7698	01/12/03	950		30	10 ACCOUNTING NEW YORK	
7902	FORD	ANALYST	7566	01/12/03	3000		20	10 ACCOUNTING NEW YORK	
7934	MILLER	CLERK	7782	02/01/23	1300		10	10 ACCOUNTING NEW YORK	
7969	SMITH	CLERK	7902	08/12/17	800		20	40 OPERATIONS BOSTON	
7489	ALLEN	SALESMAN	7698	01/02/20	1600	300	30	40 OPERATIONS BOSTON	
7900	JAMES	CLERK	7698	01/12/03	950		30	10 ACCOUNTING NEW YORK	
7902	FORD	ANALYST	7566	01/12/03	3000		20	10 ACCOUNTING NEW YORK	
7934	MILLER	CLERK	7782	02/01/23	1300		10	10 ACCOUNTING NEW YORK	
7969	SMITH	CLERK	7902	08/12/17	800		20	40 OPERATIONS BOSTON	
7489	ALLEN	SALESMAN	7698	01/02/20	1600	300	30	40 OPERATIONS BOSTON	
7521	WARD	SALESMAN	7698	01/02/22	1250	500	30	40 OPERATIONS BOSTON	
7566	JONES	MANAGER	7698	01/04/02	2975		20	40 OPERATIONS BOSTON	
7654	MARTIN	SALESMAN	7698	01/09/20	1250	1400	30	40 OPERATIONS BOSTON	
7698	BLAKE	MANAGER	7698	01/05/01	2850		30	40 OPERATIONS BOSTON	
7702	CLARK	MANAGER	7698	01/06/09	2450		10	40 OPERATIONS BOSTON	
7788	SCOTT	ANALYST	7566	07/04/19	3000		20	40 OPERATIONS BOSTON	
7839	KING	PRESIDENT		01/11/17	5000		10	40 OPERATIONS BOSTON	
7844	TURNER	SALESMAN	7698	01/09/00	1500	0	30	40 OPERATIONS BOSTON	
7876	ADAMS	CLERK	7788	07/05/23	1100		20	40 OPERATIONS BOSTON	
7900	JAMES	CLERK	7698	01/12/03	950		30	40 OPERATIONS BOSTON	
7902	FORD	ANALYST	7566	01/12/03	3000		20	40 OPERATIONS BOSTON	
7934	MILLER	CLERK	7782	02/01/23	1300		10	40 OPERATIONS BOSTON	
7969	SMITH	CLERK	7902	08/12/17	800		20	40 OPERATIONS BOSTON	
7489	ALLEN	SALESMAN	7698	01/02/20	1600	300	30	40 OPERATIONS BOSTON	
7521	WARD	SALESMAN	7698	01/02/22	1250	500	30	40 OPERATIONS BOSTON	
7566	JONES	MANAGER	7698	01/04/02	2975		20	40 OPERATIONS BOSTON	
7654	MARTIN	SALESMAN	7698	01/09/20	1250	1400	30	40 OPERATIONS BOSTON	

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

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DEPTNO DNAME	LOC
7698	BLAKE	MANAGER	7698	01/05/01	2850		30	30 SALES CHICAGO	
7782	CLARK	MANAGER	7698	01/06/09	2450		10	30 SALES CHICAGO	
7788	SCOTT	ANALYST	7566	07/04/19	3000		20	30 SALES CHICAGO	
7839	KING	PRESIDENT		01/11/17	5000		10	30 SALES CHICAGO	
7844	TURNER	SALESMAN	7698	01/09/00	1500	0	30	30 SALES CHICAGO	
7876	ADAMS	CLERK	7788	07/05/23	1100		20	30 SALES CHICAGO	
7900	JAMES	CLERK	7698	01/12/03	950		30	30 SALES CHICAGO	
7902	FORD	ANALYST	7566	01/12/03	3000		20	30 SALES CHICAGO	
7934	MILLER	CLERK	7782	02/01/23	1300		10	30 SALES CHICAGO	
7969	SMITH	CLERK	7902	08/12/17	800		20	40 OPERATIONS BOSTON	
7489	ALLEN	SALESMAN	7698	01/02/20	1600	300	30	40 OPERATIONS BOSTON	
7521	WARD	SALESMAN	7698	01/02/22	1250	500	30	40 OPERATIONS BOSTON	
7566	JONES	MANAGER	7698	01/04/02	2975		20	40 OPERATIONS BOSTON	
7654	MARTIN	SALESMAN	7698	01/09/20	1250	1400	30	40 OPERATIONS BOSTON	
7698	BLAKE	MANAGER	7698	01/05/01	2850		30	40 OPERATIONS BOSTON	
7702	CLARK	MANAGER	7698	01/06/09	2450		10	40 OPERATIONS BOSTON	
7788	SCOTT	ANALYST	7566	07/04/19	3000		20	40 OPERATIONS BOSTON	
7839	KING	PRESIDENT		01/11/17	5000		10	40 OPERATIONS BOSTON	
7844	TURNER	SALESMAN	7698	01/09/00	1500	0	30	40 OPERATIONS BOSTON	
7876	ADAMS	CLERK	7788	07/05/23	1100		20	40 OPERATIONS BOSTON	
7900	JAMES	CLERK	7698	01/12/03	950		30	40 OPERATIONS BOSTON	
7902	FORD	ANALYST	7566	01/12/03	3000		20	40 OPERATIONS BOSTON	
7934	MILLER	CLERK	7782	02/01/23	1300		10	40 OPERATIONS BOSTON	

56 개의 행이 선택되었습니다.

Cross join

- ❖ Cross Join의 결과 얻어지는 컬럼의 수는 사원 테이블의 컬럼의 수(8)와 부서 테이블의 컬럼의 수(3)를 더한 것이므로 11이 됩니다. 로우 수는 사원 한 명에 대해서 DEPT 테이블의 4개의 로우와 결합되기에 56개(14×4)가 됩니다.
- ❖ Cross Join의 결과를 보면 사원 테이블에 부서에 대한 상세정보가 결합되긴 했지만, 조인될 때 아무런 조건을 제시하지 않았기에 사원 한 명에 대해서 DEPT 테이블의 4개의 로우와 결합된 형태이기에 Cross Join의 결과는 아무런 의미를 갖지 못합니다.
- ❖ 조인 결과가 의미를 가지려면 조인할 때 조건을 지정해야 합니다.

1

조인을 사용하여 다중 테이블에서 데이터 표시

Cross join

- ❖ 조인 조건에 따라 조인의 종류가 결정되는데 다음은 조인의 종류를 정리한 표입니다.

종 류	설 명
Equi Join	동일 칼럼을 기준으로 조인합니다.
Non-Equi Join	동일 칼럼이 없이 다른 조건을 사용하여 조인합니다.
Outer Join	조인 조건에 만족하지 않는 행도 나타낸다.
Seif Join	한 테이블 내에서 조인합니다.

조인을 사용하여 다중 테이블에서 데이터 표시

Equi join

- ❖ EQUI JOIN은 가장 많이 사용하는 조인 방법으로서 조인 대상이 되는 두 테이블에서 공통적으로 존재하는 컬럼의 값이 일치되는 행을 연결하여 결과를 생성하는 조인 방법입니다.
- ❖ 다음은 사원 정보를 출력할 때 각 사원들이 소속된 부서의 상세 정보를 출력하기 위해서 두 개의 테이블을 조인한 예입니다.

예	<pre>SELECT * FROM EMP, DEPT WHERE EMP.DEPTNO = DEPT.DEPTNO;</pre>
---	--

- ❖ 사원(EMP) 테이블과 부서(DEPT) 테이블의 공통 컬럼인 DEPTNO의 값이 일치(=)되는 조건을 WHERE 절에 기술하여 사용하였습니다.
- ❖ 테이블을 조인하려면 일치되는 공통 컬럼을 사용해야 한다고 하였습니다. 컬럼의 이름이 같게 되면 혼동이 오기 때문에 컬럼 이름 앞에 테이블 이름을 기술합니다.

1

조인을 사용하여 다중 테이블에서 데이터 표시

Equi join

- ❖ 다음은 두 테이블을 조인한 결과입니다.
 결과를 살펴보면 다음과 같이 부서 번호를 기준으로 같은 값을 가진 학생 테이블의 컬럼과 부서 테이블의 컬럼이 결합됩니다.

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

```
SQL> SELECT *
2 FROM EMP, DEPT
3 WHERE EMP.DEPTNO=DEPT.DEPTNO;
```

EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DEPTNO	DNAME	LOC
7782	CLARK	MANAGER	7839	81/06/09	2450		10	10	ACCOUNTING	NEW YORK
7839	KING	PRESIDENT		81/11/17	5000		10	10	ACCOUNTING	NEW YORK
7934	MILLER	CLERK	7782	82/01/23	1300		10	10	ACCOUNTING	NEW YORK
7566	JONES	MANAGER	7839	81/04/02	2975		20	20	RESEARCH	DALLAS
7902	FORD	ANALYST	7566	81/12/03	3000		20	20	RESEARCH	DALLAS
7876	ADAMS	CLERK	7788	87/05/23	1100		20	20	RESEARCH	DALLAS
7369	SMITH	CLERK	7902	80/12/17	800		20	20	RESEARCH	DALLAS
7788	SCOTT	ANALYST	7566	87/04/19	3000		20	20	RESEARCH	DALLAS
7521	WARD	SALESMAN	7698	81/02/22	1250	500	30	30	SALES	CHICAGO
7844	TURNER	SALESMAN	7698	81/09/08	1500	0	30	30	SALES	CHICAGO
7499	ALLEN	SALESMAN	7698	81/02/20	1600	300	30	30	SALES	CHICAGO
EMPNO	ENAME	JOB	MGR	HIREDATE	SAL	COMM	DEPTNO	DEPTNO	DNAME	LOC
7900	JAMES	CLERK	7698	81/12/03	950		30	30	SALES	CHICAGO
7698	BLAKE	MANAGER	7839	81/05/01	2850		30	30	SALES	CHICAGO
7654	MARTIN	SALESMAN	7698	81/09/28	1250	1400	30	30	SALES	CHICAGO

14 개의 행이 선택되었습니다.

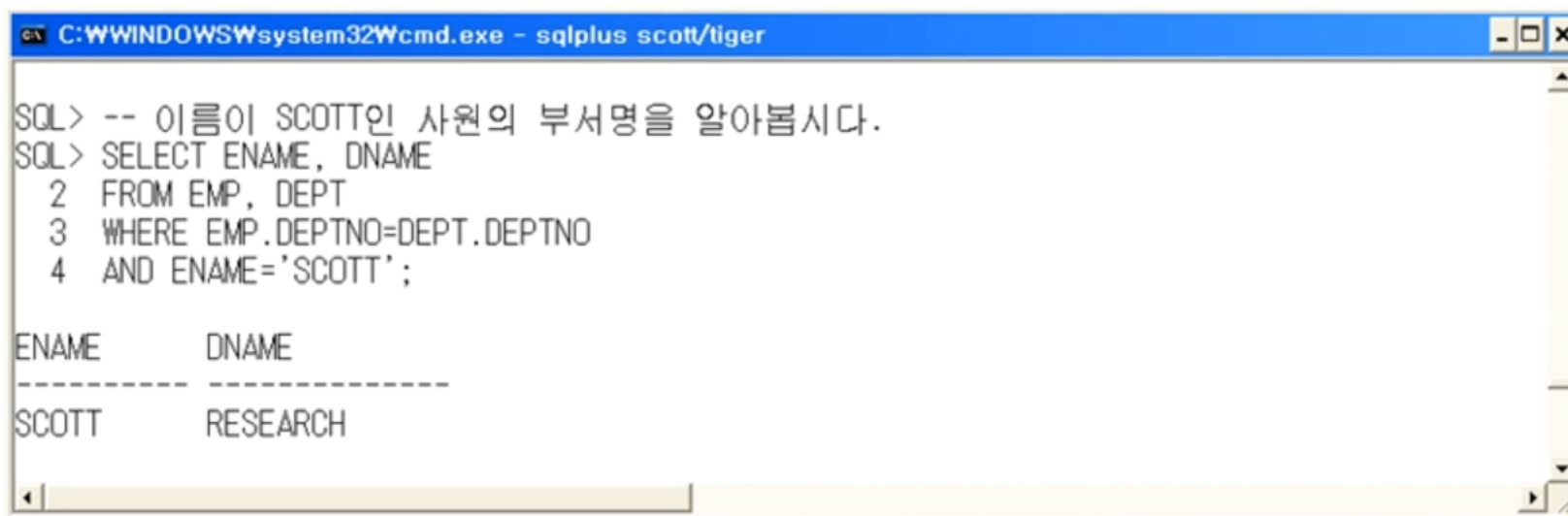
1

조인을 사용하여 다중 테이블에서 데이터 표시

Equi join에 and 연산하기

❖ 이름이 SCOTT인 사람의 부서명을 출력해봅시다.

예	<pre>SELECT ENAME, DNAME FROM EMP, DEPT WHERE EMP.DEPTNO=DEPT.DEPTNO AND ENAME='SCOTT';</pre>
---	---



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

SQL> -- 이름이 SCOTT인 사원의 부서명을 알아봅시다.
SQL> SELECT ENAME, DNAME
      2  FROM EMP, DEPT
      3  WHERE EMP.DEPTNO=DEPT.DEPTNO
      4  AND ENAME='SCOTT';

ENAME      DNAME
-----
SCOTT      RESEARCH
```

1

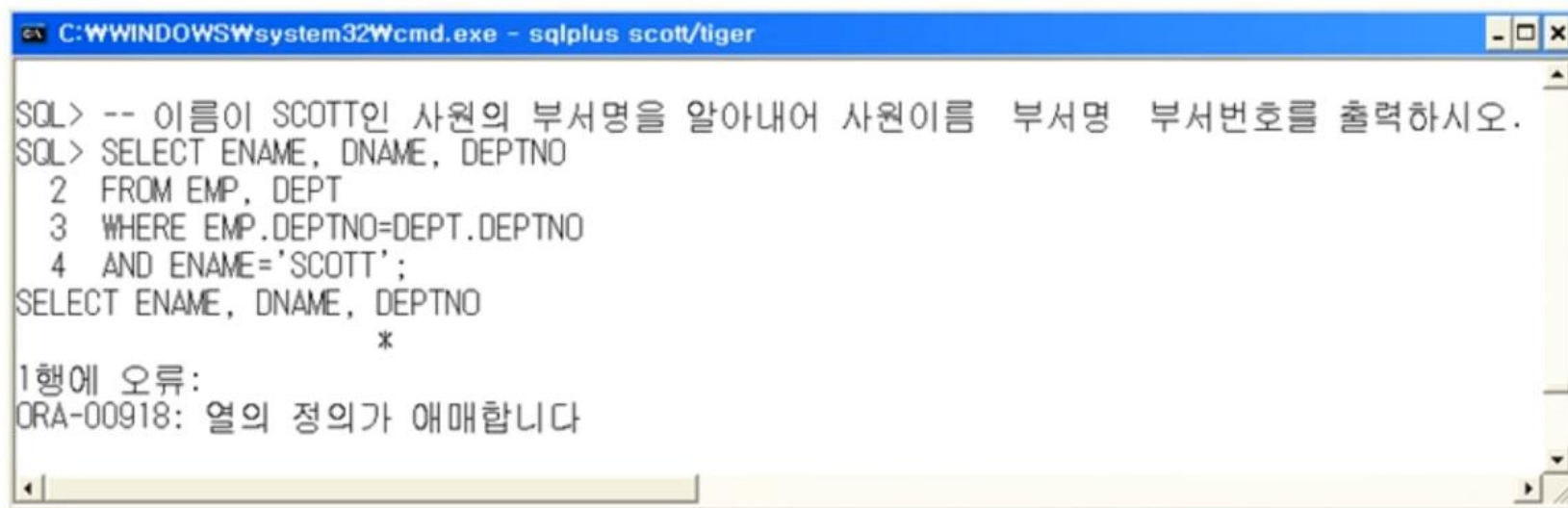
조인을 사용하여 다중 테이블에서 데이터 표시

컬럼명의 모호성 해결

- ❖ 두 테이블에 동일한 이름의 컬럼을 사용하면 어느 테이블 소속인지 불분명하기에 애매모호한 상태라는 오류 메시지가 출력됩니다.

예

```
SELECT ENAME, DNAME, DEPTNO  
FROM EMP, DEPT  
WHERE EMP.DEPTNO = DEPT.DEPTNO  
AND ENAME='SCOTT';
```



```
C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger  
SQL> -- 이름이 SCOTT인 사원의 부서명을 알아내어 사원이름 부서명 부서번호를 출력하시오.  
SQL> SELECT ENAME, DNAME, DEPTNO  
2 FROM EMP, DEPT  
3 WHERE EMP.DEPTNO=DEPT.DEPTNO  
4 AND ENAME='SCOTT';  
SELECT ENAME, DNAME, DEPTNO  
*  
1행에 오류:  
ORA-00918: 열의 정의가 애매합니다
```

1

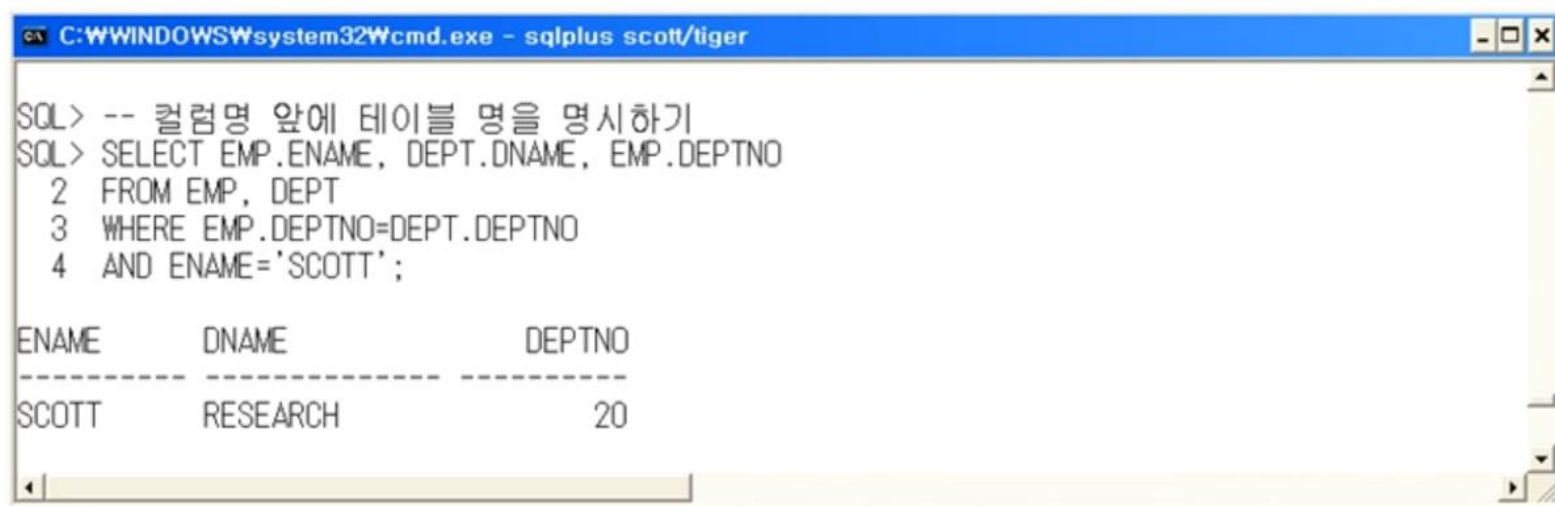
조인을 사용하여 다중 테이블에서 데이터 표시

컬럼명의 모호성 해결

- ❖ 이러한 문제를 해결하기 위한 방법이 있어야 합니다.
이렇게 동일한 이름의 컬럼은 컬럼 명 앞에 테이블 명을 명시적으로 기술함으로써 컬럼이 어느 테이블 소속인지 구분할 수 있게 됩니다.

예

```
SELECT EMP.ENAME, DEPT.DNAME, EMP.DEPTNO  
FROM EMP, DEPT  
WHERE EMP.DEPTNO=DEPT.DEPTNO  
AND ENAME='SCOTT';
```



```
C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger  
SQL> -- 컬럼명 앞에 테이블 명을 명시하기  
SQL> SELECT EMP.ENAME, DEPT.DNAME, EMP.DEPTNO  
2 FROM EMP, DEPT  
3 WHERE EMP.DEPTNO=DEPT.DEPTNO  
4 AND ENAME='SCOTT';  
  
ENAME      DNAME      DEPTNO  
-----  
SCOTT      RESEARCH      20
```


1

조인을 사용하여 다중 테이블에서 데이터 표시

테이블에 별칭 부여하기

- ❖ 테이블 이름에 별칭을 붙이려면 FROM 절 다음에 테이블 이름을 명시하고 공백을 둔 다음에 별칭을 지정하면 됩니다.

예

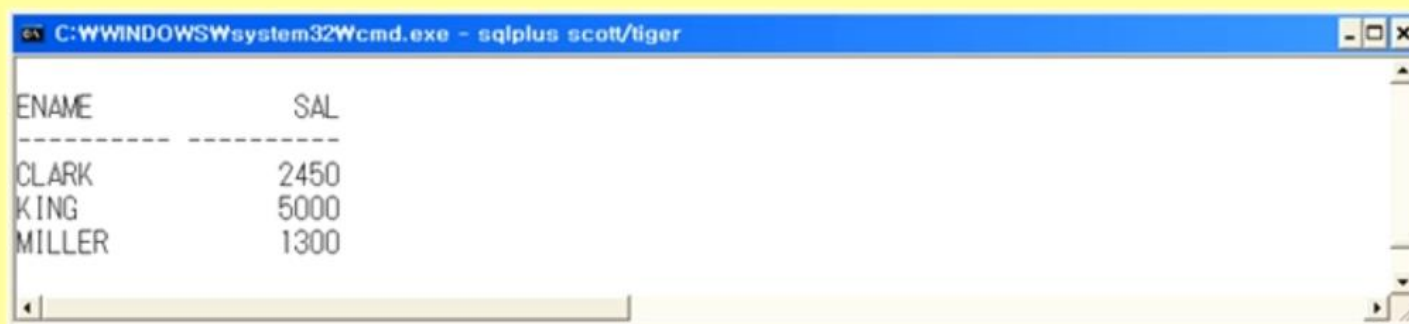


```
SELECT E.ENAME, D.DNAME, E.DEPTNO, D.DEPTNO  
FROM EMP E, DEPT D  
WHERE E.DEPTNO = D.DEPTNO  
AND E.ENAME='SCOTT';
```

1 실습.

아래와 같이 출력하시오.

1. 뉴욕에서 근무하는 사원의 이름과 급여를 출력하시오.



The screenshot shows a SQL*Plus window titled "C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger". The output displays a table with two columns: ENAME and SAL. The data rows are CLARK with salary 2450, KING with salary 5000, and MILLER with salary 1300.

ENAME	SAL
CLARK	2450
KING	5000
MILLER	1300

2. ACCOUNTING 부서 소속 사원의 이름과 입사일을 출력하시오.



The screenshot shows a SQL*Plus window titled "C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger". The output displays a table with two columns: ENAME and HIREDATE. The data rows are CLARK with hire date 81/06/09, KING with hire date 81/11/17, and MILLER with hire date 82/01/23.

ENAME	HIREDATE
CLARK	81/06/09
KING	81/11/17
MILLER	82/01/23

```
select emp.ename, emp.sal from emp, dept where emp.deptno = dept.deptno  
and dept.loc = 'NEW YORK';
```


1 실습.

아래와 같이 출력하시오.

3. 직급이 MANAGER인 사원의 이름, 부서명을 출력하시오.



ENAME	DNAME
CLARK	ACCOUNTING
JONES	RESEARCH
BLAKE	SALES

```
select emp.ename, dept.dname from emp, dept where emp.deptno = dept.deptno  
and emp.job='MANAGER'
```

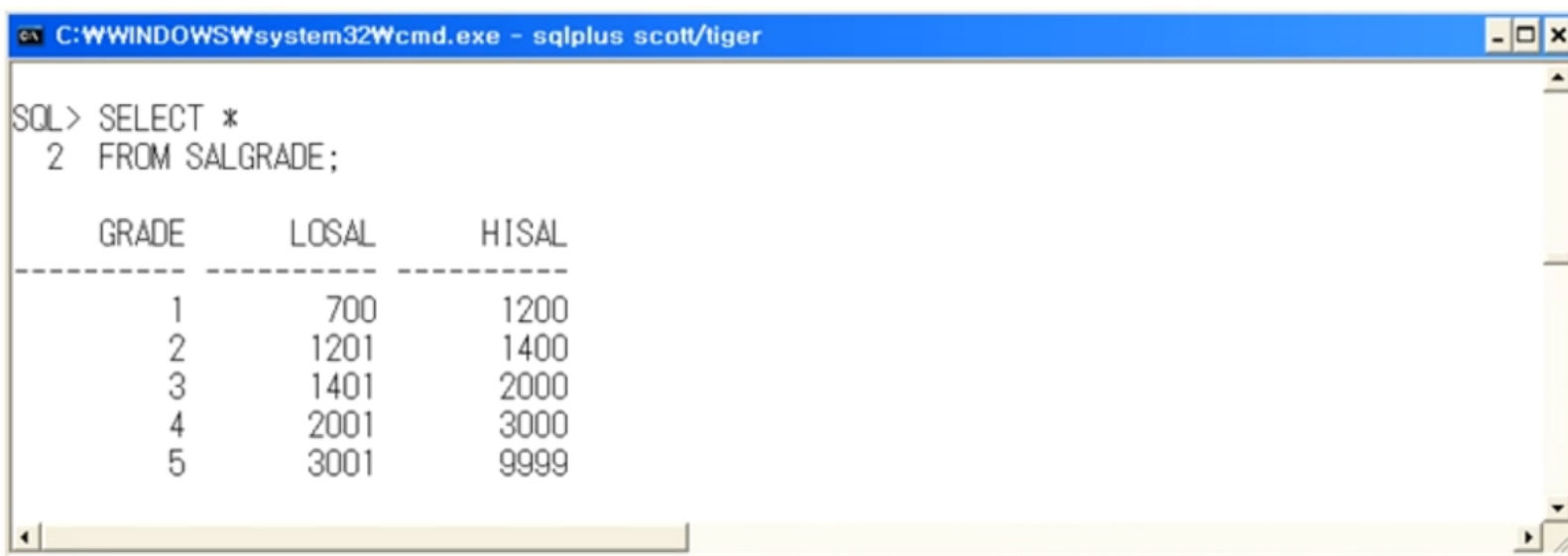
1 조인을 사용하여 다중 테이블에서 데이터 표시

Non-equi join

- ❖ Non-Equi Join은 조인 조건에 특정 범위 내에 있는지를 조사하기 위해서 WHERE 절에 조인 조건을 = 연산자 이외의 비교 연산자를 사용합니다.
- ❖ Non-Equi Join을 학습하기 전에 급여 등급 테이블(SALGRADE)을 살펴보겠습니다.

예

```
SELECT * FROM SALGRADE;
```



The screenshot shows a SQL*Plus window titled 'C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger'. The prompt 'SQL>' is followed by the command 'SELECT * FROM SALGRADE;'. The output is a table with three columns: GRADE, LOSAL, and HISAL. The data is as follows:

GRADE	LOSAL	HISAL
1	700	1200
2	1201	1400
3	1401	2000
4	2001	3000
5	3001	9999

조인을 사용하여 다중 테이블에서 데이터 표시

Non-equi join

- ❖ 급여 등급 테이블(salgrade)에는 급여에 대한 등급을 다음과 같이 나누어 놓았습니다.
- ❖ 급여의 등급은 총 5등급으로 나누어져 있으며,
1등급은 급여가 700부터 1200 사이이고,
2등급은 1201부터 1400 사이이고,
3등급은 1401부터 2000 사이이고,
4등급은 2001부터 3000사이이고,
5등급이면 3001부터 9999사이입니다.
- ❖ 급여 등급을 5개로 나누어 놓은 salgrade에서 정보를 얻어 와서 각 사원의 급여 등급을 지정해보도록 합시다.
이를 위해서 사원(emp) 테이블과 급여 등급(salgrade) 테이블을 조인하도록 합시다.
다음은 사원의 급여가 몇 등급인지 살펴보는 예제입니다.


1

조인을 사용하여 다중 테이블에서 데이터 표시

Non-equi join

예

```
SELECT ENAME, SAL, GRADE  
FROM EMP, SALGRADE  
WHERE SAL BETWEEN LOSAL AND HISAL;
```



```
C:\WINDOWS\system32\cmd.exe - sqlplus scott/tiger  
SQL> SELECT ENAME, SAL, GRADE  
2 FROM EMP, SALGRADE  
3 WHERE SAL BETWEEN LOSAL AND HISAL;  
  
ENAME          SAL      GRADE  
-----  
SMITH          800        1  
JAMES          950        1  
ADAMS         1100        1  
WARD          1250        2  
MARTIN         1250        2  
MILLER         1300        2  
TURNER         1500        3  
ALLEN          1600        3  
CLARK          2450        4  
BLAKE          2850        4  
JONES          2975        4  
  
ENAME          SAL      GRADE  
-----  
SCOTT          3000        4  
FORD           3000        4  
KING           5000        5  
  
14 개의 행이 선택되었습니다.
```


THANK YOU