

〈2주차 실습〉

SQL 실습

Database Programming



Oracle – SQL 실습



MySQL – SQL 실습



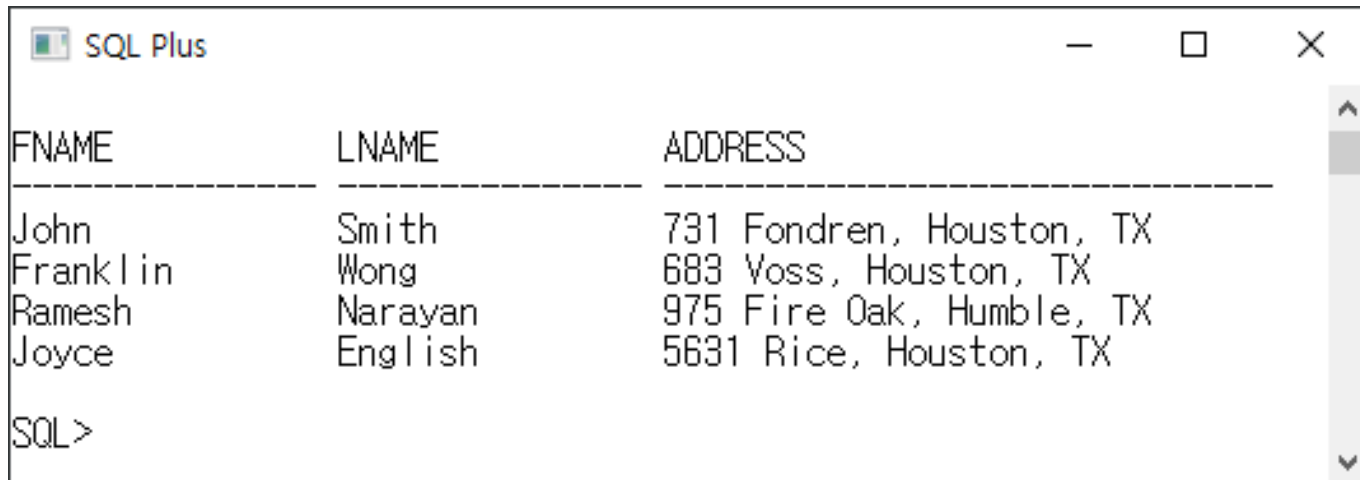
MS SQL Server – SQL 실습



실습 과제

- 1> 'Research' 부서에서 근무하는 모든 사원의 이름(Fname, Lname)과 주소(Address)를 검색하라.

```
SELECT Fname, Lname, Address
FROM EMPLOYEE, DEPARTMENT
WHERE Dname = 'Research'
AND Dnumber = Dno;
```



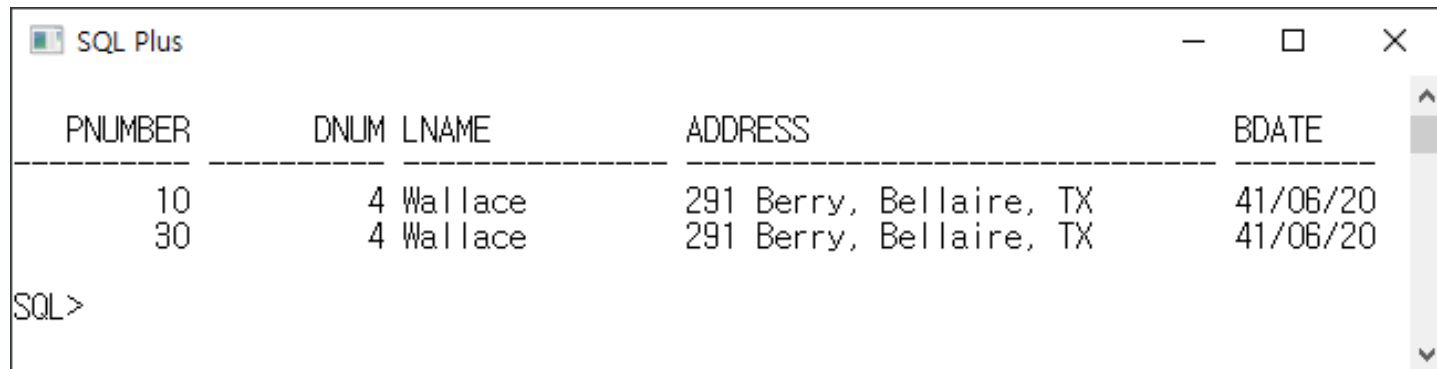
The screenshot shows a window titled "SQL Plus" with a table of query results. The table has three columns: FNAME, LNAME, and ADDRESS. The results are as follows:

FNAME	LNAME	ADDRESS
John	Smith	731 Fondren, Houston, TX
Franklin	Wong	683 Voss, Houston, TX
Ramesh	Narayan	975 Fire Oak, Humble, TX
Joyce	English	5631 Rice, Houston, TX

SQL>

- 2> 'Stafford'에 위치한 모든 프로젝트에 대해서 프로젝트 번호(Pnumber), 담당부서 번호(Dnum), 부서관리자의 성(Lname), 주소(Address), 생년월일(Bdate)로 검색하라.

```
SELECT Pnumber, Dnum, Lname, Address, Bdate
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE Dnum = Dnumber
AND Mgr_Ssn = Ssn
AND Plocation = 'Stafford';
```

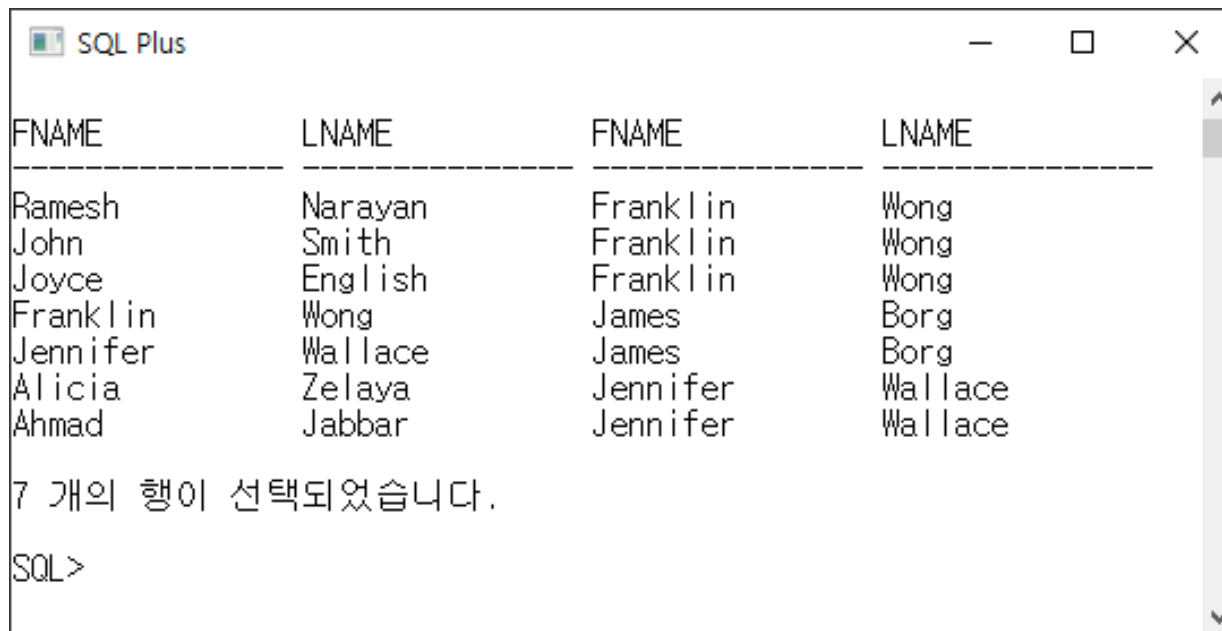


The screenshot shows a SQL Plus window titled "SQL Plus" with standard window controls (minimize, maximize, close). The query results are displayed in a table with five columns: PNUMBER, DNUM, LNAME, ADDRESS, and BDATE. The results show two rows of data for project 10 and 30, both managed by Wallace at the same address. The prompt "SQL>" is visible at the bottom left.

PNUMBER	DNUM	LNAME	ADDRESS	BDATE
10	4	Wallace	291 Berry, Bellaire, TX	41/06/20
30	4	Wallace	291 Berry, Bellaire, TX	41/06/20

3> 각 사원에 대해 사원의 이름(Fname)과 성(Lname), 직속 상사의 이름(Fname)과 성(Lname)을 검색하라.

```
SELECT E.Fname, E.Lname, S.Fname, S.Lname
FROM EMPLOYEE E, EMPLOYEE S
WHERE E.Super_Ssn = S.Ssn;
```



The screenshot shows a window titled "SQL Plus" with a standard Windows-style title bar (minimize, maximize, close buttons). The window displays the results of the SQL query. The results are presented in a table with four columns: FNAME, LNAME, FNAME, and LNAME. The first two columns represent the employee's name, and the next two columns represent their supervisor's name. There are 7 rows of data. Below the table, a message states "7 개의 행이 선택되었습니다." (7 rows selected). The prompt "SQL>" is visible at the bottom left.

FNAME	LNAME	FNAME	LNAME
Ramesh	Narayan	Franklin	Wong
John	Smith	Franklin	Wong
Joyce	English	Franklin	Wong
Franklin	Wong	James	Borg
Jennifer	Wallace	James	Borg
Alicia	Zelaya	Jennifer	Wallace
Ahmad	Jabbar	Jennifer	Wallace

7 개의 행이 선택되었습니다.

SQL>

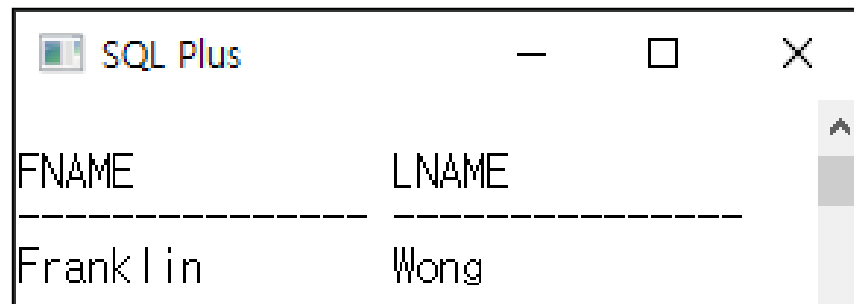
4> 1950년대에 태어난 모든 사원을 검색하라. (LIKE 연산자 사용)

- Oracle

```
SELECT Fname, Lname  
FROM EMPLOYEE  
WHERE Bdate LIKE '5_____';
```

- MySQL, MS SQL Server

```
SELECT Fname, Lname  
FROM EMPLOYEE  
WHERE Bdate LIKE '195_____';
```

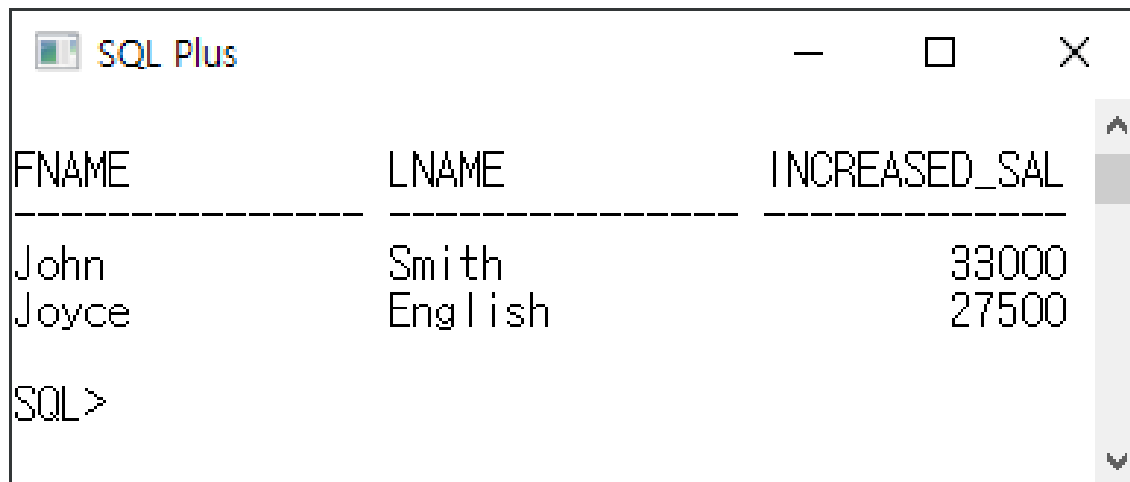


The screenshot shows a window titled "SQL Plus" with standard Windows window controls (minimize, maximize, close). The window displays the results of a query in a two-column format. The first column is labeled "FNAME" and the second is labeled "LNAME". Below the headers, there is a single row of data with the values "Franklin" and "Wong".

FNAME	LNAME
Franklin	Wong

5> 'ProductX' 프로젝트에 참여하는 모든 사원의 급여를 10% 올린 경우의 급여를 구하라.

```
SELECT E.Fname, E.Lname, 1.1*E.Salary AS Increased_sal
FROM EMPLOYEE E, WORKS_ON W, PROJECT P
WHERE E.Ssn = W.Essn
AND W.Pno = P.Pnumber
AND P.Pname = 'ProductX';
```



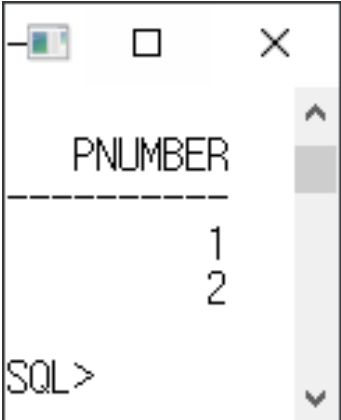
The screenshot shows a window titled "SQL Plus" with standard window controls (minimize, maximize, close). It displays the output of the SQL query. The results are presented in a table with three columns: FNAME, LNAME, and INCREASED_SAL. The data rows show John Smith with an increased salary of 33000 and Joyce English with an increased salary of 27500. The prompt "SQL>" is visible at the bottom.

FNAME	LNAME	INCREASED_SAL
John	Smith	33000
Joyce	English	27500

SQL>

- 6> 일반 직원 또는 관리자의 성(Lname)이 'Smith'인 사원이 참여하는 모든 프로젝트의 프로젝트 번호(Pnumber)를 검색하라. (UNION 사용)

```
SELECT DISTINCT Pnumber
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE Dnum = Dnumber
AND Mgr_ssn = Ssn
AND Lname = 'Smith'
UNION
SELECT DISTINCT Pnumber
FROM PROJECT, WORKS_ON, EMPLOYEE
WHERE Pnumber = Pno
AND Essn = Ssn
AND Lname = 'Smith';
```



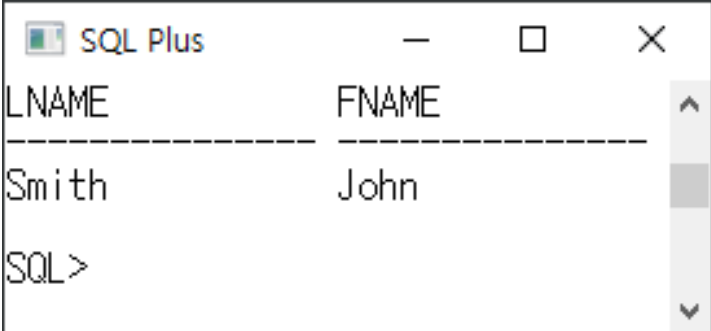
PNUMBER
1
2

SQL>

7> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (IN/중첩 질의/TO_CHAR 사용)

- Oracle

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE E
WHERE E.Ssn IN (SELECT E.Ssn
                FROM DEPENDENT D
                WHERE TO_CHAR(E.Bdate,'MM') = TO_CHAR(D.Bdate,'MM')
                AND E.Sex = D.Sex);
```



The screenshot shows a window titled "SQL Plus" with a table of query results. The table has two columns: "LNAME" and "FNAME". The first row of data shows "Smith" under LNAME and "John" under FNAME. Below the table is the prompt "SQL>".

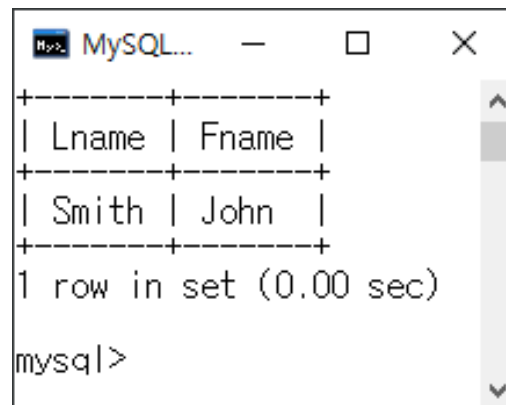
LNAME	FNAME
Smith	John

SQL>

7> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (IN/중첩 질의/DATE_FORMAT 사용)

- MySQL

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E
WHERE E.Ssn IN (SELECT E.Ssn
                FROM DEPENDENT AS D
                WHERE DATE_FORMAT(E.Bdate,'%m') = DATE_FORMAT (D.Bdate,'%m')
                AND E.Sex = D.Sex);
```



The screenshot shows a MySQL command window titled 'MySQL...'. It displays the output of the SQL query, which is a table with two columns: 'Lname' and 'Fname'. The first row of data shows 'Smith' and 'John'. Below the table, it indicates '1 row in set (0.00 sec)'. The prompt 'mysql>' is visible at the bottom.

Lname	Fname
Smith	John

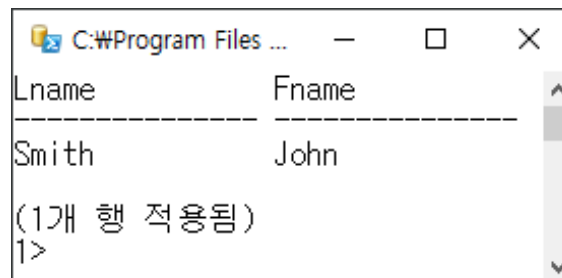
1 row in set (0.00 sec)

mysql>

7> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (IN/중첩 질의/MONTH 사용)

- MS SQL Server

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E
WHERE E.Ssn IN (SELECT E.Ssn
                FROM DEPENDENT AS D
                WHERE E.Ssn=D.Essn
                AND E.Sex=D.Sex
                AND MONTH(E.Bdate) = MONTH(D.Bdate));
GO
```



C:\Program Files ...

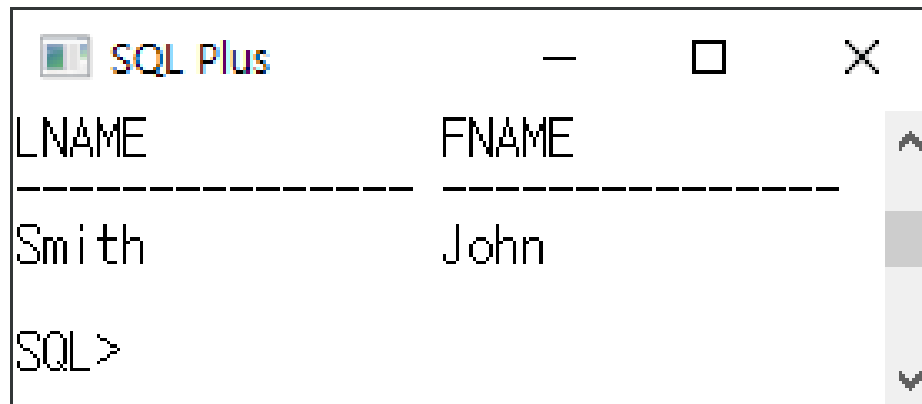
Lname	Fname
Smith	John

(1개 행 적용됨)
1>

8> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (단일 블록 질의/TO_CHAR 사용)

- Oracle

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE E, DEPENDENT D
WHERE E.Ssn=D.Essn
AND E.Sex=D.Sex
AND TO_CHAR(E.Bdate,'MM') = TO_CHAR(D.Bdate,'MM');
```



The screenshot shows a window titled "SQL Plus" with a standard Windows-style title bar (minimize, maximize, close buttons). The window displays the results of a SQL query in a two-column format. The columns are labeled "LNAME" and "FNAME", separated by a dashed line. Below the header, the first row of data shows "Smith" under LNAME and "John" under FNAME. At the bottom of the window, the prompt "SQL>" is visible. A vertical scrollbar is on the right side of the results area.

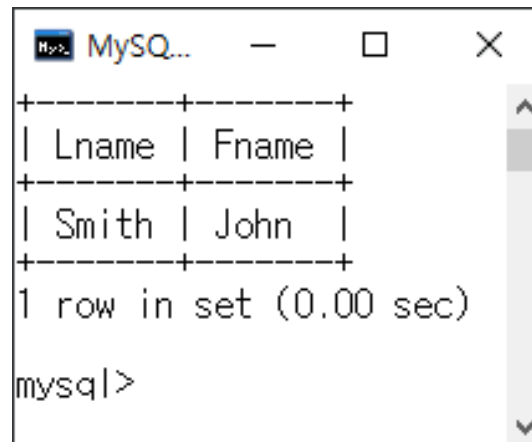
LNAME	FNAME
Smith	John

SQL>

8> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 직원들의 성(Lname), 이름(Fname)을 검색하라. (단일 블록 질의/DATE_FORMAT 사용)

- MySQL

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E, DEPENDENT AS D
WHERE E.Ssn=D.Essn
AND E.Sex=D.Sex
AND DATE_FORMAT(E.Bdate,'%m') = DATE_FORMAT(D.Bdate,'%m');
```



The screenshot shows a MySQL command window with the title 'MySQL...'. It displays the output of the SQL query, which is a table with two columns: 'Lname' and 'Fname'. The table contains one row with the values 'Smith' and 'John'. Below the table, it says '1 row in set (0.00 sec)'. The prompt 'mysql>' is visible at the bottom.

Lname	Fname
Smith	John

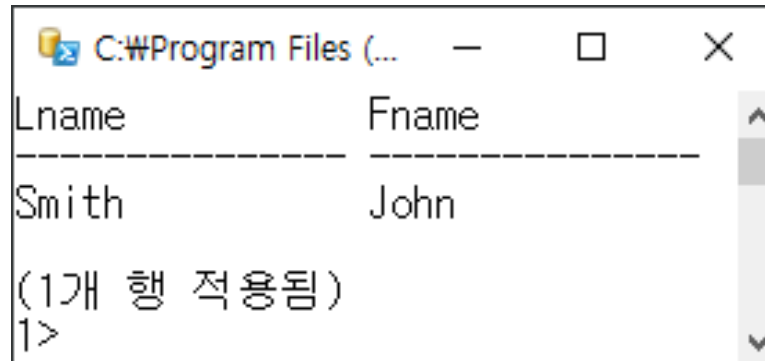
1 row in set (0.00 sec)

mysql>

8> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (단일 블록 질의/MONTH 사용)

- MS SQL Server

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E, DEPENDENT AS D
WHERE E.Ssn=D.Essn
AND E.Sex=D.Sex
AND MONTH(E.Bdate) = MONTH(D.Bdate);
GO
```



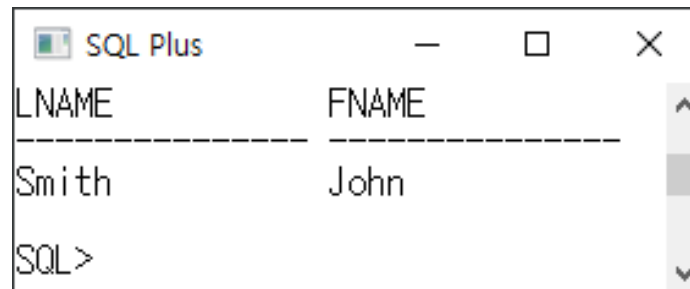
Lname	Fname
Smith	John

(1개 행 적용됨)
1>

9> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (EXISTS/TO_CHAR 사용)

- Oracle

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE E
WHERE EXISTS (SELECT *
              FROM DEPENDENT D
              WHERE E.Ssn=D.Essn
              AND E.Sex=D.Sex
              AND TO_CHAR(E.Bdate,'MM') = TO_CHAR(D.Bdate,'MM'));
```



The screenshot shows a window titled "SQL Plus" with standard window controls. It displays the results of the SQL query in a table format. The table has two columns: "LNAME" and "FNAME". The first row of data shows "Smith" under LNAME and "John" under FNAME. Below the table, the prompt "SQL>" is visible. A vertical scrollbar is on the right side of the table.

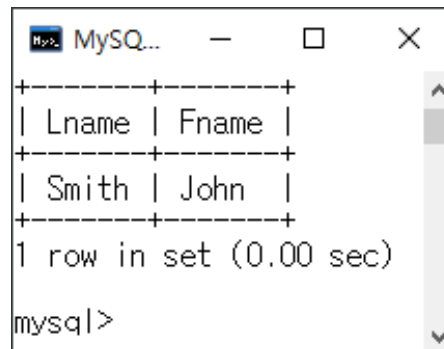
LNAME	FNAME
Smith	John

SQL>

9> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (EXISTS/DATE_FORMAT 사용)

- MySQL

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E
WHERE EXISTS (SELECT *
              FROM DEPENDENT AS D
              WHERE E.Ssn=D.Essn
              AND E.Sex=D.Sex
              AND DATE_FORMAT(E.Bdate,'%m') = DATE_FORMAT(D.Bdate,'%m'));
```



The screenshot shows a MySQL command window titled 'MySQL...'. It displays the output of the SQL query, which is a table with two columns: 'Lname' and 'Fname'. The table contains one row with the values 'Smith' and 'John'. Below the table, it says '1 row in set (0.00 sec)'. The prompt 'mysql>' is visible at the bottom.

Lname	Fname
Smith	John

1 row in set (0.00 sec)

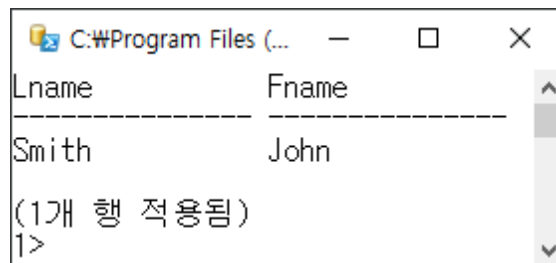
mysql>

9> 사원과 부양가족의 출생 월과 성별(Sex)이 서로 같은 사원들의 성(Lname), 이름(Fname)을 검색하라. (EXISTS/MONTH 사용)

- MS SQL Server

```
SELECT E.Lname, E.Fname
FROM EMPLOYEE AS E
WHERE EXISTS (SELECT *
              FROM DEPENDENT AS D
              WHERE E.Ssn=D.Essn
              AND E.Sex=D.Sex
              AND MONTH(E.Bdate) = MONTH(D.Bdate));

GO
```



A screenshot of a Microsoft SQL Server query result window. The window title is 'C:\Program Files (...)' with standard minimize, maximize, and close buttons. The results are displayed in a table with two columns: 'Lname' and 'Fname'. The first row shows 'Smith' under 'Lname' and 'John' under 'Fname'. Below the table, it indicates '(1개 행 적용됨)' (1 row affected) and a prompt '1>'.

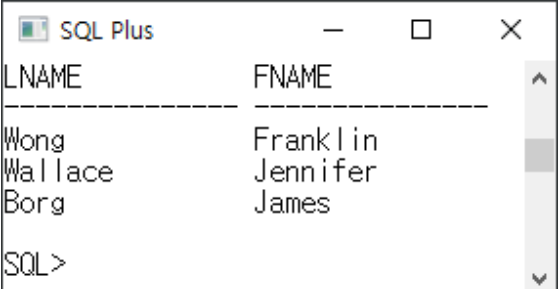
Lname	Fname
Smith	John

(1개 행 적용됨)
1>

10> 1번 부서가 담당하는 모든 프로젝트에 근무하는 직원들의 성(Lname), 이름(Fname)을 검색하라. (NOT EXISTS/MINUS 사용)

- Oracle

```
SELECT Lname, Fname
FROM EMPLOYEE
WHERE NOT EXISTS ((SELECT Pnumber
                    FROM PROJECT
                    WHERE Dnum = 1)
MINUS
(SELECT Pno
FROM WORKS_ON
WHERE Ssn=Essn));
```



The screenshot shows a window titled 'SQL Plus' with a table of results. The table has two columns, 'LNAME' and 'FNAME', separated by a dashed line. The data rows are: Wong Franklin, Wallace Jennifer, and Borg James. Below the table, the prompt 'SQL>' is visible. A vertical scrollbar is on the right side of the table.

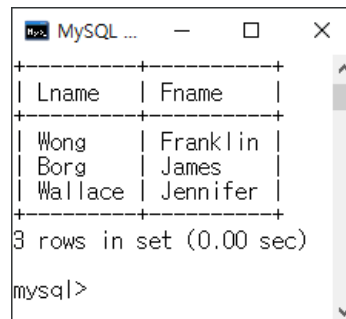
LNAME	FNAME
Wong	Franklin
Wallace	Jennifer
Borg	James

SQL>

10> 1번 부서가 담당하는 모든 프로젝트에 근무하는 직원들의 성(Lname), 이름(Fname)을 검색하라. (NOT EXISTS/NOT IN 사용)

- MySQL

```
SELECT Lname, Fname
FROM EMPLOYEE AS E
WHERE NOT EXISTS (SELECT Pnumber
                  FROM PROJECT AS P
                  WHERE Dnum = 1
                  AND P.Pnumber NOT IN (SELECT W.Pno
                                         FROM WORKS_ON W
                                         WHERE E.Ssn=W.Essn));
```



Lname	Fname
Wong	Franklin
Borg	James
Wallace	Jennifer

3 rows in set (0.00 sec)

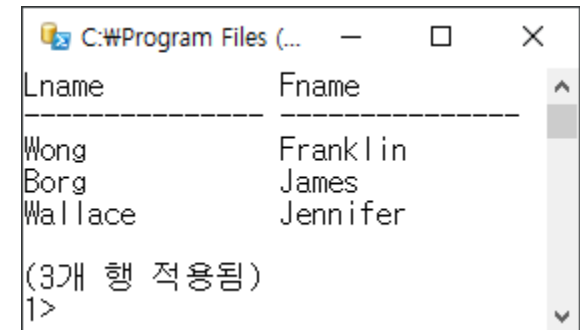
mysql>

10> 1번 부서가 담당하는 모든 프로젝트에 근무하는 직원들의 성(Lname), 이름(Fname)을 검색하라. (NOT EXISTS/EXCEPT 사용)

- MS SQL Server

```
SELECT Lname, Fname
FROM EMPLOYEE
WHERE NOT EXISTS ((SELECT Pnumber
                    FROM PROJECT
                    WHERE Dnum = 1)
EXCEPT
(SELECT Pno
 FROM WORKS_ON
 WHERE Ssn=Essn));

GO
```



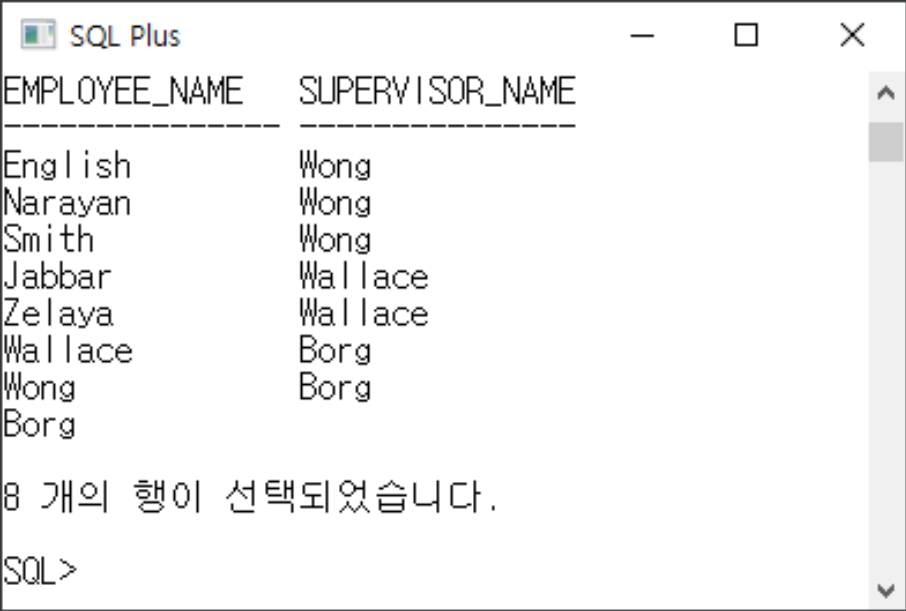
The screenshot shows a window titled 'C:\Program Files (...)' with a table of results. The table has two columns: 'Lname' and 'Fname'. The data rows are: Wong, Franklin; Borg, James; Wallace, Jennifer. Below the table, it says '(3개 행 적용됨)' and '1>'. The window has standard Windows controls (minimize, maximize, close) in the top right corner.

Lname	Fname
Wong	Franklin
Borg	James
Wallace	Jennifer

(3개 행 적용됨)
1>

11> 모든 사원의 성(Lname)과 직속상사가 있는 경우 직속상사의 성(Lname)을 함께 검색하라. (LEFT OUTER JOIN 사용)

```
SELECT E.Lname AS Employee_name, S.Lname AS Supervisor_name
FROM EMPLOYEE E LEFT OUTER JOIN EMPLOYEE S
ON E.Super_Ssn=S.Ssn;
```



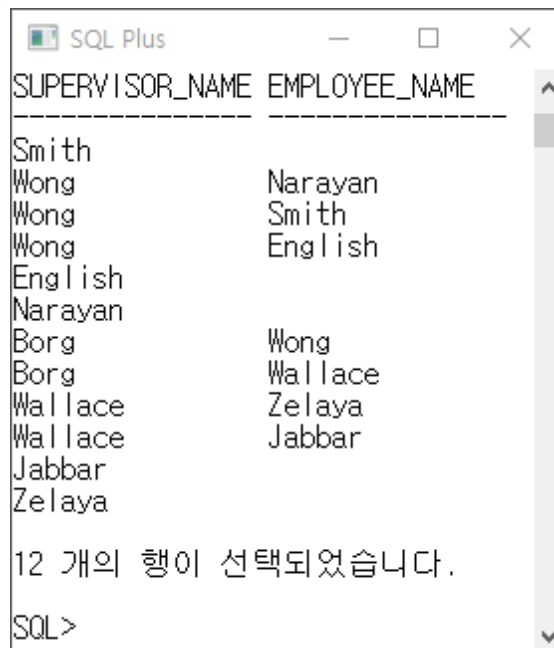
The screenshot shows a SQL Plus window titled "SQL Plus" with a table of query results. The table has two columns: "EMPLOYEE_NAME" and "SUPERVISOR_NAME". The results are as follows:

EMPLOYEE_NAME	SUPERVISOR_NAME
English	Wong
Narayan	Wong
Smith	Wong
Jabbar	Wallace
Zelaya	Wallace
Wallace	Borg
Wong	Borg
Borg	

Below the table, the text "8 개의 행이 선택되었습니다." (8 rows selected) is displayed, followed by the "SQL>" prompt.

12> 모든 사원의 성(Lname)과 직속부하가 있는 경우 직속부하의 성(Lname)을 함께 검색하라. (RIGHT OUTER JOIN 사용)

```
SELECT S.Lname AS Supervisor_name, E.Lname AS Employee_name
FROM EMPLOYEE E RIGHT OUTER JOIN EMPLOYEE S
ON E.Super_Ssn=S.Ssn;
```



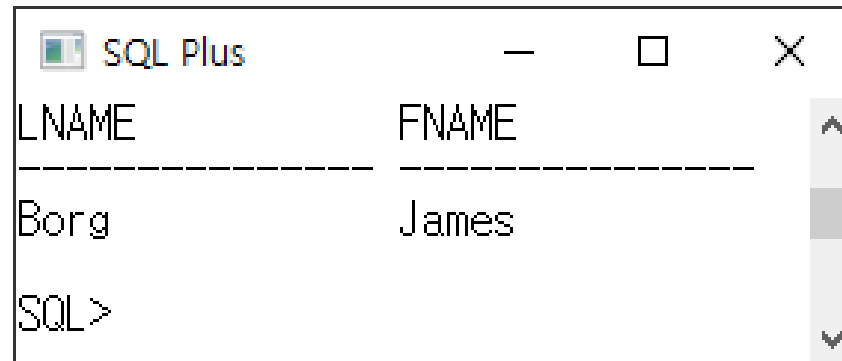
SUPERVISOR_NAME	EMPLOYEE_NAME
Smith	
Wong	Narayan
Wong	Smith
Wong	English
English	
Narayan	
Borg	Wong
Borg	Wallace
Wallace	Zelaya
Wallace	Jabbar
Jabbar	
Zelaya	

12 개의 행이 선택되었습니다.

SQL>

13> 상사가 없는 모든 사원의 성(Lname), 이름(Fname)을 검색하라.
(IS NULL 사용)

```
SELECT Lname, Fname  
FROM EMPLOYEE  
WHERE Super_ssn IS NULL;
```



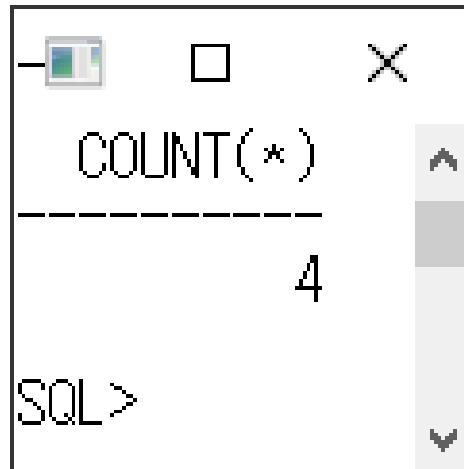
The screenshot shows a window titled "SQL Plus" with standard window controls. It displays the results of the SQL query in a table format. The table has two columns: "LNAME" and "FNAME", separated by a dashed line. The first row of data shows "Borg" under LNAME and "James" under FNAME. Below the table, the prompt "SQL>" is visible. A vertical scrollbar is on the right side of the table area.

LNAME	FNAME
Borg	James

SQL>

14> 'Research' 부서에서 근무하는 총 직원수를 검색하라. (집계함수 사용)

```
SELECT COUNT(*)  
FROM EMPLOYEE, DEPARTMENT  
WHERE DNO=DNUMBER  
AND DNAME='Research';
```



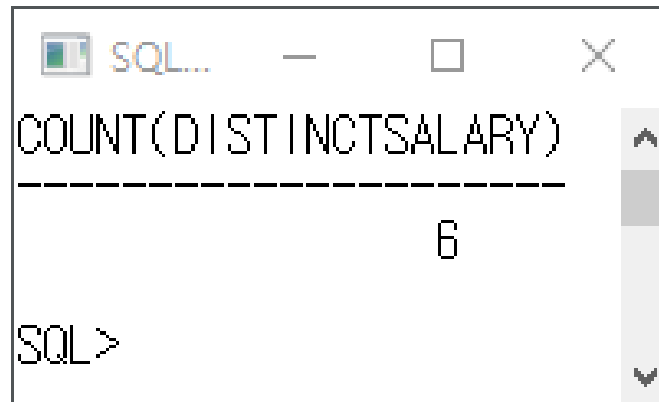
A screenshot of a SQL query result window. The window has a title bar with a minus sign, a maximize button, and a close button. The main content area displays the result of the query: a single row with the value '4' under the column header 'COUNT(*)'. The text is centered. At the bottom left, there is a prompt 'SQL>'. On the right side, there is a vertical scrollbar with up and down arrows.

COUNT(*)
4

SQL>

15> 서로 다른 급여들의 개수를 검색하라. (집계함수 사용)

```
SELECT COUNT(DISTINCT Salary)
FROM EMPLOYEE;
```



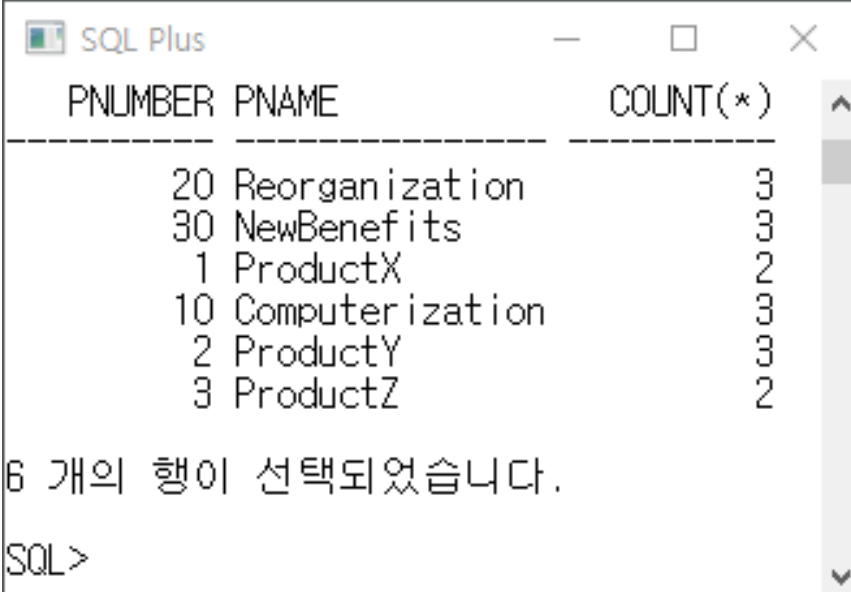
A screenshot of a SQL query window titled "SQL...". The window displays the following text:

```
COUNT(DISTINCTSALARY)
-----
6
```

At the bottom of the window, the prompt "SQL>" is visible. The window has standard Windows-style window controls (minimize, maximize, close) in the title bar.

16> 각 프로젝트에 대해서 프로젝트 번호(Pnumber), 프로젝트 이름(Pname), 그 프로젝트에서 근무하는 사원들의 수를 검색하라. (집계함수 사용)

```
SELECT Pnumber, Pname, COUNT(*)  
FROM PROJECT, WORKS_ON  
WHERE Pnumber = Pno  
GROUP BY Pnumber, Pname;
```



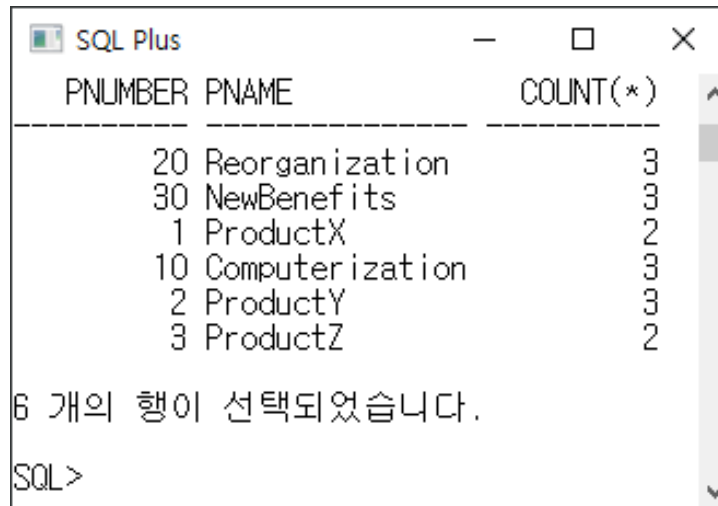
The screenshot shows a SQL Plus window titled "SQL Plus" with a table of results. The table has three columns: PNUMBER, PNAME, and COUNT(*). The data is as follows:

PNUMBER	PNAME	COUNT(*)
20	Reorganization	3
30	NewBenefits	3
1	ProductX	2
10	Computerization	3
2	ProductY	3
3	ProductZ	2

Below the table, the text "6 개의 행이 선택되었습니다." (6 rows selected) is displayed, followed by the prompt "SQL>".

- 17> 두 명 이상의 사원이 근무하는 각 프로젝트에 대해서 프로젝트 번호(Pnumber), 프로젝트 이름(Pname), 프로젝트에서 근무하는 사원의 수를 검색하라. (집계함수 사용)

```
SELECT Pnumber, Pname, COUNT(*)  
FROM PROJECT, WORKS_ON  
WHERE Pnumber = Pno  
GROUP BY Pnumber, Pname  
HAVING COUNT(*) >=2;
```



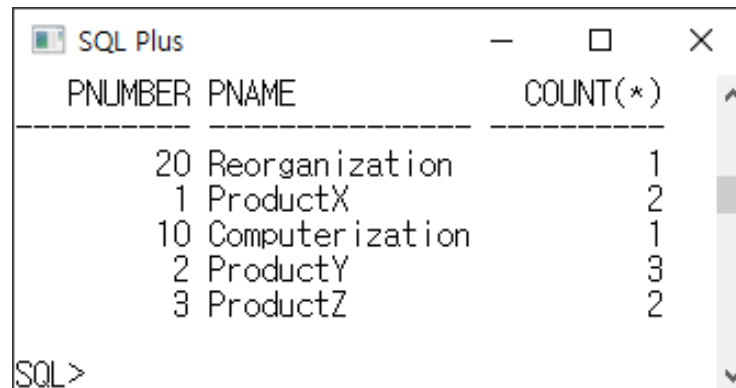
The screenshot shows a SQL Plus window titled 'SQL Plus' with standard window controls. It displays the results of the SQL query. The results are presented in a table with three columns: PNUMBER, PNAME, and COUNT(*). There are six rows of data. Below the table, a message states '6 개의 행이 선택되었습니다.' (6 rows selected). The prompt 'SQL>' is visible at the bottom.

PNUMBER	PNAME	COUNT(*)
20	Reorganization	3
30	NewBenefits	3
1	ProductX	2
10	Computerization	3
2	ProductY	3
3	ProductZ	2

6 개의 행이 선택되었습니다.
SQL>

18> 5번 부서의 직원들이 참여하는 프로젝트의 프로젝트 번호(Pnumber), 프로젝트 이름(Pname), 직원수를 검색하라. (집계함수 사용)

```
SELECT Pnumber, Pname, COUNT(*)  
FROM PROJECT, WORKS_ON, EMPLOYEE  
WHERE Pnumber = Pno  
AND Ssn = Essn  
AND Dno = 5  
GROUP BY Pnumber, Pname;
```



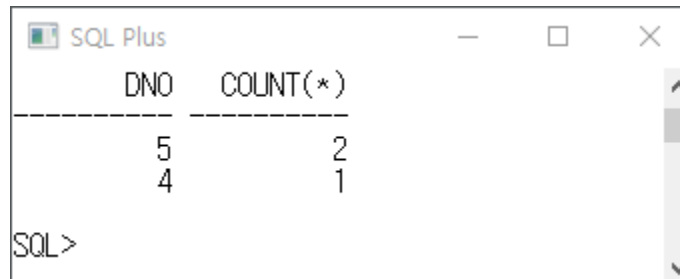
The screenshot shows a window titled "SQL Plus" with a table of results. The table has three columns: PNUMBER, PNAME, and COUNT(*). The data is as follows:

PNUMBER	PNAME	COUNT(*)
20	Reorganization	1
1	ProductX	2
10	Computerization	1
2	ProductY	3
3	ProductZ	2

The prompt "SQL>" is visible at the bottom left of the window.

19> 2명 이상의 사원이 근무하는 각 부서에 대해서 만 부서 번호(Dno)와 35,000달러가 넘는 급여를 받는 사원의 수를 검색하라. (중첩 질의 사용)

```
SELECT Dno, COUNT(*)  
FROM EMPLOYEE  
WHERE Salary > 35000  
AND Dno IN (SELECT Dno  
            FROM EMPLOYEE  
            GROUP BY Dno  
            HAVING COUNT(*) >=2)  
GROUP BY Dno;
```



A screenshot of a SQL Plus window titled "SQL Plus". It displays the results of the SQL query in a table with two columns: "DNO" and "COUNT(*)". The table has two rows of data: DNO 5 with a count of 2, and DNO 4 with a count of 1. Below the table, the prompt "SQL>" is visible. The window has standard Windows-style controls (minimize, maximize, close) in the top right corner.


DNO	COUNT(*)
5	2
4	1

SQL>

20> 2명 이상의 사원이 근무하는 각 부서에 대해서 만 부서 번호(Dno)와 35,000달러가 넘는 급여를 받는 사원의 수를 검색하라. (WITH 사용)

```
WITH BIGDEPTS (Dno) AS (SELECT Dno
                           FROM EMPLOYEE
                           GROUP BY Dno
                           HAVING COUNT(*) >= 2)

SELECT Dno, COUNT(*)
FROM EMPLOYEE
WHERE Salary >= 35000
AND Dno IN (SELECT Dno
             FROM BIGDEPTS)
GROUP BY Dno;
```



A screenshot of a SQL Plus window titled 'SQL Plus'. It displays the results of a query in a table with two columns: 'DNO' and 'COUNT(*)'. The table has two rows of data. Below the table, the prompt 'SQL>' is visible. The window has standard OS controls (minimize, maximize, close) in the top right corner.

DNO	COUNT(*)
5	2
4	1

21> EMPLOYEE 릴레이션에 새로운 사원의 정보를 삽입하라.

```
INSERT INTO EMPLOYEE
```

```
VALUES ('Richard', 'K', 'Marini', '653298653', '1962-12-30',
```

```
'98 Oak Forest, Katy, TX', 'M', 37000, '653298653', 4);
```

SQL Plus

FNAME	M LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard	K Marini	653298653	62/12/30	98 Oak Forest, Katy, TX	M	37000	653298653	4
John	B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	40000	888665555	5
Alicia	J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A English	453453453	72/07/31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

9 개의 행이 선택되었습니다.

SQL>

22> EMPLOYEE 릴레이션에 새로운 사원의 일부 애트리뷰트의 값만 삽입하라.

```
INSERT INTO EMPLOYEE(Fname, Lname, Dno, Ssn)
VALUES ('Richard', 'Marini', 4, '764309762');
```

SQL Plus

FNAME	M	LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard		K Marini	653298653	62/12/30	98 Oak Forest, Katy, TX	M	37000	653298653	4
Richard		Marini	764309762						4
John		B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin		T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	40000	888665555	5
Alicia		J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer		S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh		K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce		A English	453453453	72/07/31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad		V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James		E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

10 개의 행이 선택되었습니다.

SQL>

23> 10번 프로젝트를 담당하는 부서의 위치는 'Bellaire', 부서번호를 5로 수정하라.

```
UPDATE PROJECT
SET Plocation = 'Bellaire', Dnum = 5
WHERE Pnumber = 10;
```

SQL Plus

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Stafford	4
Reorganization	20	Houston	1
NewBenefits	30	Stafford	4

6 개의 행이 선택되었습니다.

SQL>



SQL Plus

PNAME	PNUMBER	PLOCATION	DNUM
ProductX	1	Bellaire	5
ProductY	2	Sugarland	5
ProductZ	3	Houston	5
Computerization	10	Bellaire	5
Reorganization	20	Houston	1
NewBenefits	30	Stafford	4

6 개의 행이 선택되었습니다.

SQL>

24> 'Research' 부서에 근무하는 모든 사원의 급여를 10% 인상하라.

```
UPDATE EMPLOYEE
SET Salary = Salary * 1.1
WHERE Dno = 5;
```

SQL Plus

FNAME	M LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard	K Marini	653298653	62/12/30	98 Oak Forest, Katy, TX	M	37000	653298653	4
Richard	Marini	764309762						4
John	B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	40000	888665555	5
Alicia	J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A English	453453453	72/07/31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

10 개의 행이 선택되었습니다.

SQL>



SQL Plus

FNAME	M LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard	K Marini	653298653	62/12/30	98 Oak Forest, Katy, TX	M	37000	653298653	4
Richard	Marini	764309762						4
John	B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	33000	333445555	5
Franklin	T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	44000	888665555	5
Alicia	J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	41800	333445555	5
Joyce	A English	453453453	72/07/31	5631 Rice, Houston, TX	F	27500	333445555	5
Ahmad	V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

10 개의 행이 선택되었습니다.

SQL>

25> EMPLOYEE 릴레이션에서 사원 번호(Ssn)가 '653298653'인 사원의 정보를 삭제하라.

```
DELETE FROM EMPLOYEE
WHERE Ssn = '653298653';
```

SQL Plus

FNAME	M	LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard	K	Marini	653298653	62/12/30	98 Oak Forest, Katy, TX	M	37000	653298653	4
Richard		Marini	764309762						4
John	B	Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	33000	333445555	5
Franklin	T	Wong	333445555	55/12/08	683 Voss, Houston, TX	M	44000	888665555	5
Alicia	J	Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	41800	333445555	5
Joyce	A	English	453453453	72/07/31	5631 Rice, Houston, TX	F	27500	333445555	5
Ahmad	V	Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

10 개의 행이 선택되었습니다.

SQL>



SQL Plus

FNAME	M	LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard		Marini	764309762						4
John	B	Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	33000	333445555	5
Franklin	T	Wong	333445555	55/12/08	683 Voss, Houston, TX	M	44000	888665555	5
Alicia	J	Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	41800	333445555	5
Joyce	A	English	453453453	72/07/31	5631 Rice, Houston, TX	F	27500	333445555	5
Ahmad	V	Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

9 개의 행이 선택되었습니다.

SQL>

26> EMPLOYEE 릴레이션에서 사원의 성(Lname)이 'Marini'인 사원의 정보를 삭제하라.

```
DELETE FROM EMPLOYEE
WHERE Lname = 'Marini';
```

SQL Plus

FNAME	M LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
Richard	Marini	764309762						4
John	B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	33000	333445555	5
Franklin	T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	44000	888665555	5
Alicia	J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	41800	333445555	5
Joyce	A English	453453453	72/07/31	5631 Rice, Houston, TX	F	27500	333445555	5
Ahmad	V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

9 개의 행이 선택되었습니다.

SQL>



SQL Plus

FNAME	M LNAME	SSN	BDATE	ADDRESS	S	SALARY	SUPER_SSN	DNO
John	B Smith	123456789	65/01/09	731 Fondren, Houston, TX	M	33000	333445555	5
Franklin	T Wong	333445555	55/12/08	683 Voss, Houston, TX	M	44000	888665555	5
Alicia	J Zelaya	999887777	68/01/19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S Wallace	987654321	41/06/20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K Narayan	666884444	62/09/15	975 Fire Oak, Humble, TX	M	41800	333445555	5
Joyce	A English	453453453	72/07/31	5631 Rice, Houston, TX	F	27500	333445555	5
Ahmad	V Jabbar	987987987	69/03/29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E Borg	888665555	37/11/10	450 Stone, Houston, TX	M	55000		1

8 개의 행이 선택되었습니다.

SQL>

- 릴레이션 스키마

ITEM (ItemID, Description, PurchaseDate, Store, City, Quantity, LocalCurrencyAmount, ExchangeRate)

SHIPMENT (ShipmentID, ShipperName, ShipperInvoiceNumber, DepartureDate, ArrivalDate, InsuredValue)

SHIPMENT_ITEM (ShipmentID, ShipmentItemID, ItemID, Value)

ITEM

Column Name	Type	Key	Required	Remarks
ItemID	Integer	Primary Key	Yes	Surrogate Key
Description	Character (255)	No	Yes	Use Varchar
PurchaseDate	Date	No	Yes	
Store	Character (50)	No	Yes	
City	Character (35)	No	Yes	
Quantity	Integer	No	Yes	
LocalCurrencyAmount	Number (18,2)	No	Yes	
ExchangeRate	Number (12,6)	No	Yes	

- 릴레이션 스키마

SHIPMENT

Column Name	Type	Key	Required	Remarks
ShipmentID	Integer	Primary Key	Yes	Surrogate Key
ShipperName	Character (35)	No	Yes	
ShipperInvoiceNumber	Integer	No	Yes	
DepartureDate	Date	No	No	
ArrivalDate	Date	No	No	
InsuredValue	Number (12,2)	No	No	

SHIPMENT_ITEM

Column Name	Type	Key	Required	Remarks
ShipmentID	Integer	Primary Key, Foreign Key	Yes	REF: SHIPMENT
ShipmentItemID	Integer	Primary Key	Yes	Sequential number, but <i>not</i> a surrogate key
ItemID	Integer	Foreign Key	Yes	REF: ITEM
Value	Number (12,2)	No	Yes	

• 테이블

ITEM

ItemID	Description	PurchaseDate	Store	City	Quantity	LocalCurrencyAmount	ExchangeRate
1	QE Dining Set	07-Apr-15	Eastern Treasures	Manila	2	403405	0.01774
2	Willow Serving Dishes	15-Jul-15	Jade Antiques	Singapore	75	102	0.5903
3	Large Bureau	17-Jul-15	Eastern Sales	Singapore	8	2000	0.5903
4	Brass Lamps	20-Jul-15	Jade Antiques	Singapore	40	50	0.5903

SHIPMENT

ShipmentID	ShipperName	ShipperInvoiceNumber	DepartureDate	ArrivalDate	InsuredValue
1	ABC Trans-Oceanic	2008651	10-Dec-14	15-Mar-15	\$15,000.00
2	ABC Trans-Oceanic	2009012	10-Jan-15	20-Mar-15	\$12,000.00
3	Worldwide	49100300	05-May-15	17-Jun-15	\$20,000.00
4	International	399400	02-Jun-15	17-Jul-15	\$17,500.00
5	Worldwide	84899440	10-Jul-15	28-Jul-15	\$25,000.00
6	International	488955	05-Aug-15	11-Sep-15	\$18,000.00

SHIPMENT_ITEM

ShipmentID	ShipmentItemID	ItemID	Value
3	1	1	\$15,000.00
4	1	4	\$1,200.00
4	2	3	\$9,500.00
4	3	2	\$4,500.00

- 제공한 릴레이션 스키마, 인스턴스를 참고하여 Oracle, MySQL, MS SQL Server에서 데이터베이스를 구현하고 다음의 질의 실행 결과를 제출하시오.

Q1> InsuredValue가 \$10,000.00 이상인 모든 선적의 ShipmentID, ShipperName, ShipperInvoiceNumber를 검색하라.

Q2> 선적자 이름이 'AB'로 시작하는 모든 선적의 ShipmentID, ShipperName, ShipperInvoiceNumber를 검색하라.

Q3> DepartmentDate와 ArrivalDate가 MM/DD/YY 형식이라고 가정하자. December에 출발한 모든 선적의 ShipmentID, ShipperName, ShipperInvoiceNumber, ArrivalDate를 검색하라.

Q4> DepartmentDate와 ArrivalDate가 MM/DD/YY 형식이라고 가정하자. 매월 10일에 출발한 모든 선적의 ShipmentID, ShipperName, ShipperInvoiceNumber, ArrivalDate를 검색하라.

Q5> InsuredValue의 최대값, 최소값과 평균을 검색하라.

Q6> ITEM의 모든 행에 대하여 ItemID, Description, Store, LocalCurrencyAmt에 ExchangeRate를 곱한 USCurrencyAmount를 검색하라.

Q7> City와 Store별로 구매 건수를 검색하라.

Q8> 가격이 \$1,000.00 이상인 상품을 포함하는 모든 선적의 ShipperName, DepartureDate를 검색하라. 중첩 질의를 사용하고, ShipperName은 오름차순, DepartureDate는 내림차순으로 정렬하라.

Q9> Singapore에서 구입한 상품을 포함하는 모든 선적의 ShipperName, DepartureDate를 중복을 제거하여 검색하라. 중첩 질의와 조인을 사용하고, ShipperName은 오름차순, DepartureDate는 내림차순으로 정렬하라.

Q10> Shipment 릴레이션에 아래의 데이터를 추가로 삽입하라.

ShipmentID	ShipperInvoice Name	ShipperInvoice Number	DepartureDate	ArrivalDate	InsuredValue
7	Worldwide	98765432	10-Sep-20	14-Sep-20	\$23,000.00

Q11> Shipment 릴레이션에 앞서 삽입한 Shipment의 정보를 아래와 같이 수정하라.

ShipmentID	ShipperInvoice Name	ShipperInvoice Number	DepartureDate	ArrivalDate	InsuredValue
7	Worldwide	12345678	12-Sep-20	14-Sep-20	\$24,000.00

Q12> Shipment 릴레이션에 앞서 삽입한 Shipment의 정보를 삭제하라.

- 제출 방식 : E-Class를 통하여 제출
- 제출 내용 : spool file(각 DBMS에서 실행한 로그 파일)
- 제출 형식 : 학번_이름_주차
 - Ex) 학번_홍길동_2주차.zip (3개의 로그 파일을 압축하여 제출)
- 제출 기한 : 수업 시작 시간으로 부터 24시간 이내 제출
 - 제출 기한 위반 시 감점 기준
 - 지각 제출 시 과제 점수에서 40% 감점
 - 1일 초과 당 10% 추가 감점 (단, 7일 이후 제출 불가)