데이터베이스 프로그래밍

JOIN

학습 목표

- JOIN을 이용하여 여러 테이블로부터 데이터 검색 가능
- EQUI JOIN 및 NON-EQUI JOIN을 이용하여, 하나 이상의 테이블로 접근 가능
- OUTER JOIN이나 SELF JOIN을 이용하여 JOIN 조건에 맞지 않는 데이터나, 자신의 테이블 접근 가능
- 그 밖에 여러 JOIN을 이용하여 다양한 SQL 문 작성 가능
- JOIN ~ USING, JOIN ~ ON 등 ANSI SQL을 활용한 조인 사용 가능

학습할 내용

- JOIN의 개념
- ORACLE JOIN 구문 및 활용 방법
  - ☞ EQUI JOIN
  - ☞ NON_EQUI JOIN
  - ☞ SELF JOIN
  - ☞ OUTER JOIN
- ANSI JOIN 구문 및 활용
  - ☞ CROSS JOIN
  - ☞ NATURAL JOIN
  - ☞ USING 절을 사용한 JOIN
  - ☞ ON 절을 사용한 JOIN
  - ☞ LEFT OUTER JOIN
  - ☞ RIGHT OUTER JOIN
  - ☞ FULL OUTER JOIN

JOIN이란?

- 여러 테이블의 데이터를 질의할 경우 사용
- 즉, 데이터베이스에서 여러 테이블의 데이터가 필요한 경우 사용
- 관계형 데이터베이스에서 가장 기본적이고 중요한 기능
- 관계형 데이터베이스에서는 서로 독립적인 데이터들간의 조인을 이용하여 필요시 원하는 다양한 정보 참조
- 하나 이상의 테이블이나 뷰의 데이터를 ROW로 결합하여, 어떤 테이블을 기준으로 다른 테이블에 있는 ROW를 검색하는 방법
- 일반적인 경우 Primary key(PK)나 Foreign key(FK) 값의 연관에 의해 조인 성립
- PK, FK의 관계가 없어도 논리적인 값들의 연관만으로 JOIN 작업 가능
- 해당 열에 존재하는 공통 값, 일반적으로 PK 및 FK열을 조인 조건으로 사용하여 한 테이블의 행을 다른 테이블의 행에 조인

JOIN이란?

- **ORACLE JOIN 구문**

```
SELECT table1.column, table2.column, ...
FROM table1, table2
WHERE table1.column1 = table2.column2;
```

- **설명**
  - FROM 절에 필요로 하는 테이블을 모두 적는다.
  - 컬럼 이름의 모호성을 피하기 위해(어느 테이블에 속하는지 알 수 없음)이 있을 수 있으므로 table명에 alias 사용(table명으로 직접 지칭 가능)➔ **table명 alias**로 이용(as 사용하지 않음)
  - 적절한 JOIN 조건을 WHERE 절에 부여(일반적으로 테이블 개수 – 1개의 조인 조건이 필요)
  - 일반적으로 PK와 FK 간의 = 조건이 붙는 경우가 많음

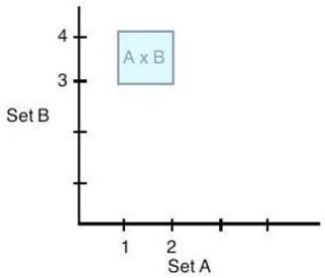
JOIN의 종류

종류		설 명
Inner Join	Equijoin	· 두 개의 테이블들간에 column 값들이 정확하게 일치하는 경우 사용 · 대부분 PK, FK의 관계를 기반으로 함 · 조건에 맞는 데이터만 조회
	Non-Equijoin	· 두 개의 테이블들간에 column 값들이 정확하게 일치하지 않는 경우 사용 · 조건에 맞는 데이터만 조회
Outer Join		· 두 개의 테이블들간에 JOIN을 걸었을 경우 JOIN 조건을 만족하지 않는 데이터들도 같이 조회하는 경우 사용 · (+)라는 연산자 사용 · 조건에 맞지 않는 데이터도 조회 예) 부서원이 없는 40, 50 부서 조회 부서 배정이 안 된 인턴사원 조회 · ERD에서 ○ 기호일 때 Outer Join일 가능성 있음
Self Join		· 두 개의 테이블 간에 JOIN을 거는 것이 아니라, 같은 테이블에 있는 행들을 JOIN하는 데 사용 · 테이블 자체가 계층구조일 때 사용 · 권장하지 않음

카티시안 곱(Cartesian Product)

데카르트 곱
(Descartes=Des+Cartes)

Ex2) $A=\{x : x\in\mathbb{R} \text{ and } 1\le x\le 2\}$ $B=\{y : y\in\mathbb{R} \text{ and } 3\le y\le 4\}$



<그림 1>

범위로 주어질 경우 <그림 1>과 같이 영역을 나타낸다.

Definition: 공집합이 아닌 두 집합 A, B에 대하여 집합 곱은 다음과 같이 정의한다.

$$A\times B=\{(a,b) : a\in A \text{ and } b\in B\}$$

a는 A집합의 원소, b는 B집합의 원소이며 $A\times B$는 순서쌍을 원소로 가지는 집합이다.

Ex) $A=\{1,2,3\}$ $B=\{4,5,6\}$

$$A\times B=\{(1,4),(1,5),(1,6),(2,4),(2,5),(2,6),(3,4),(3,5),(3,6)\}$$

여기서 순서쌍 성분은 굉장히 중요하다. 예를 들어 $(4,1)\notin A\times B$이다. 왜냐하면 $4\notin A$이고 $1\notin B$이기 때문이다.

- 카티시안 곱은 다음과 같은 경우 생성됨
  - 조인 조건을 생략한 경우
  - 조인 조건이 부적합한 경우
- 첫 번째 테이블의 모든 행이 두 번째 테이블의 모든 행이 조인되어 처리
- 카타시안 곱이 생성되지 않도록 하려면 WHERE 절에 항상 유효한 조인 조건을 지정해야 함
- A 테이블에서 얻을 수 있는 ROW 수가 n개, B 테이블에서 얻을 수 있는 ROW 수가 m개 일 때, JOIN의 조건 없이 이 두 테이블을 FROM 절에 기술할 경우, 데이터는 m X n개의 데이터 반환

사원명, 부서번호(Table:employees), 부서번호(Table:departments), 사원이 속한 부서명 조회

```
SELECT employees.first_name, employees.department_id,
  departments.department_id, departments.department_name
FROM employees, departments
ORDER BY employees.first_name;
```

결과 – 2997개의 데이터 출력

- Cartesian Product 발생 상황
- SELECT * FROM departments; -- 27개의 데이터
- SELECT * FROM employees; -- 111개의 데이터
- 카티시안 발생 이유? ➔ WHERE 절이 없으므로

EQUI JOIN

- **EQUI JOIN**

```
SELECT table1.column, table2.column, ...
FROM table1, table2
WHERE table1.column1 = table2.column2;
```

- **JOIN의 조건은 WHERE 절에 기술**

  ➔ Cartesian Product 발생 상황 해결

Cartesian Product 발생 상황 해결

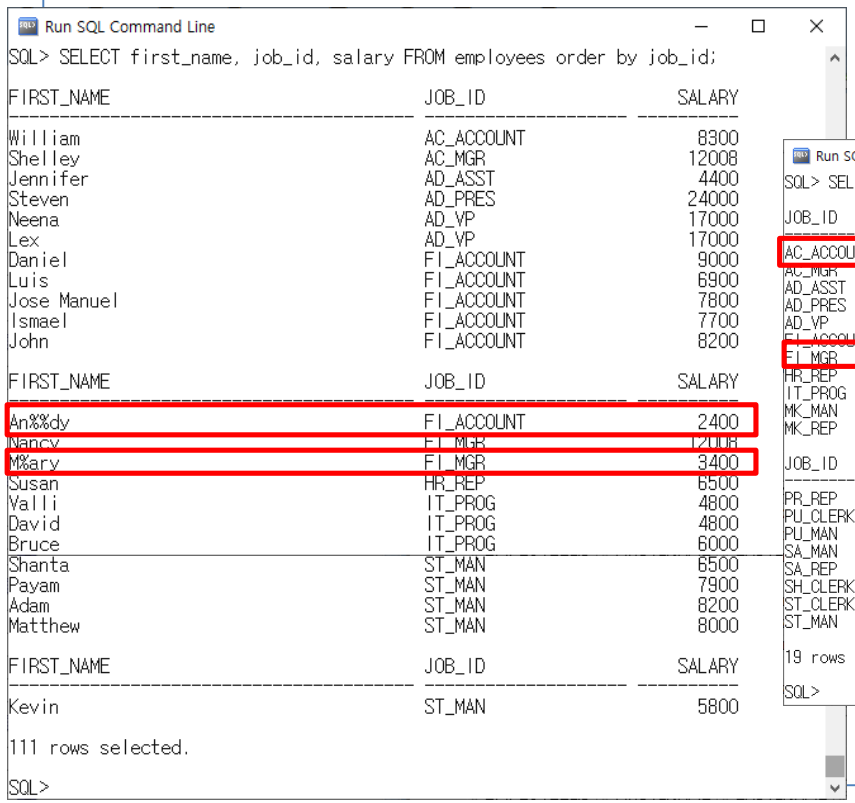
```
SELECT employees.first_name 사원명, employees.department_id 부서1,
  departments.department_id 부서2, departments.department_name 부서명
FROM employees, departments
WHERE employees.department_id=departments.department_id
ORDER BY employees.first_name;
```

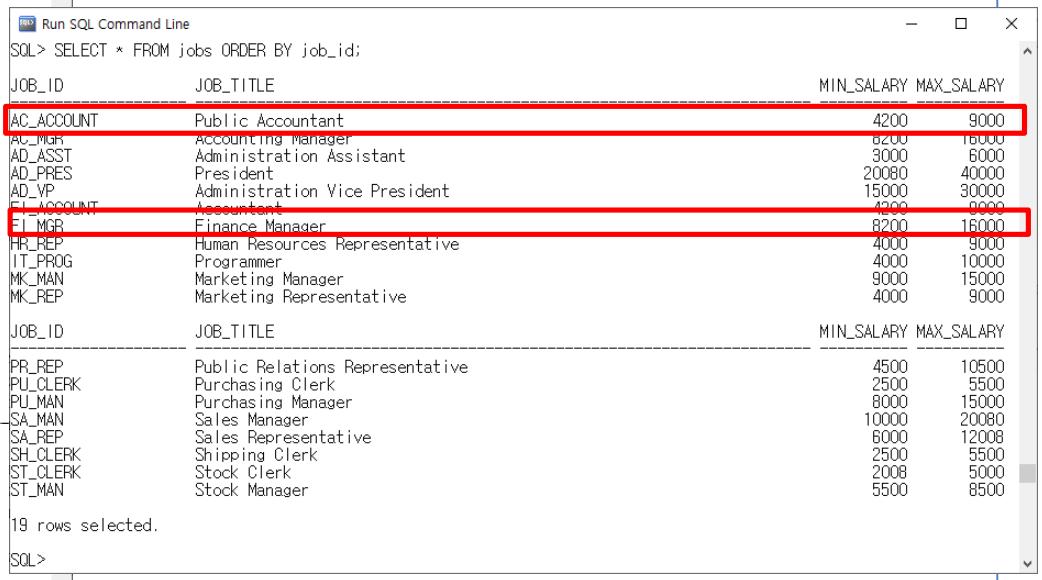
Run SQL Command Line

```
SQL>  SELECT employees.first_name 사원명, employees.department_id 부서1,
  2     departments.department_id 부서2, departments.department_name 부서명
  3   FROM employees, departments
  4   WHERE employees.department_id=departments.department_id
  5   ORDER BY employees.first_name;

사원명                                  부서1      부서2 부서명
------------------------------------ --------- --------- ------------------------------------------------------------
Adam                                        50        50 Shipping
Alana                                       50        50 Shipping
Alberto                                     80        80 Sales
Alexander                                   60        60 IT
Alexander                                   30        30 Purchasing
Alexis                                      50        50 Shipping
Susan                                       40        40 Human Resources
TJ                                          50        50 Shipping
Tayler                                      80        80 Sales
Timothy                                     50        50 Shipping
Trenna                                      50        50 Shipping
Valli                                       60        60 IT
Vance                                       50        50 Shipping
William                                     80        80 Sales
William                                    110       110 Accounting
Winston                                     50        50 Shipping

110 rows selected.

SQL>
```

NON-EQUI JOIN

- 등호연산자(=) 외의 다른 비교연산자를 이용하여 두 개 이상의 테이블을 JOIN하는 경우





급여가 최소급여와 최대급여 사이인 사원의 사원번호, 사원명, 업무코드, 급여, 업무명, 업무의 최소급여, 업무의 최대급여를 업무코드 순으로 조회

```
SELECT emp.employee_id, emp.first_name, emp.job_id, emp.salary,
  job.job_title, job.min_salary, job.max_salary
FROM employees emp, jobs job
WHERE emp.job_id = job.job_id AND
 emp.salary>=job.min_salary AND emp.salary<=job.max_salary
order by job_id;
```

emp.salary between job.min_salary AND job.max_salary

WHERE 절 없이 실행

SELECT * FROM employees; 실행

SELECT * FROM jobs; 실행

Run SQL Command Line

```
SQL> SELECT emp.employee_id, emp.first_name, emp.job_id, emp.salary,
  2    job.job_title, job.min_salary, job.max_salary
  3  FROM employees emp, jobs job
  4  WHERE emp.job_id = job.job_id AND emp.salary>=job.min_salary
  5    AND emp.salary<=job.max_salary
  6  order by job_id;
```

EMPLOYEE_ID	FIRST_NAME	JOB_ID	SALARY	JOB_TITLE	MIN_SALARY	MAX_SALARY
206	William	AC_ACCOUNT	8300	Public Accountant	4200	9000
205	Shelley	AC_MGR	12008	Accounting Manager	8200	16000
200	Jennifer	AD_ASST	4400	Administration Assistant	3000	6000
100	Steven	AD_PRES	24000	President	20080	40000
101	Neena	AD_VP	17000	Administration Vice President	15000	30000
102	Lex	AD_VP	17000	Administration Vice President	15000	30000
109	Daniel	FI_ACCOUNT	9000	Accountant	4200	9000
113	Luis	FI_ACCOUNT	6900	Accountant	4200	9000
112	Jose Manuel	FI_ACCOUNT	7800	Accountant	4200	9000
111	Ismael	FI_ACCOUNT	7700	Accountant	4200	9000
110	John	FI_ACCOUNT	8200	Accountant	4200	9000
142	Curtis	ST_CLERK	3100	Stock Clerk	2008	5000
123	Shanta	ST_MAN	6500	Stock Manager	5500	8500
122	Payam	ST_MAN	7900	Stock Manager	5500	8500
121	Adam	ST_MAN	8200	Stock Manager	5500	8500
120	Matthew	ST_MAN	8000	Stock Manager	5500	8500
124	Kevin	ST_MAN	5800	Stock Manager	5500	8500

107 rows selected.

SQL>

OUTER JOIN

- JOIN 조건을 만족하지 못하는 경우에도 모든 행들을 다 보고자 하는 경우에 사용
- (+) 연산자 사용 → 기준이 아닌 테이블
- 어느 한쪽 테이블이 기준이 되어 다른 쪽 테이블에 연결되는 조건의 만족여부에 상관없이 기준이 되는 테이블은 무조건 추출되는 JOIN
- 기준이 되는 테이블은 항상 JOIN에 성공 → 기준 테이블은 다 보여줌
- JOIN 조건을 만족하지 않는 행이라도, WHERE 절의 어느 한쪽을 기준으로 모두 보고자 하는 경우, 기준이 되는 반대쪽에 (+)를 이용하여 JOIN → 데이터가 부족한 편에 사용
- (+) 기호는 어는 한 쪽에만 위치
- 양쪽의 데이터를 다 보고자 하는 경우에는 FULL [OUTER] JOIN 이용

employees 테이블과 departments 테이블에서 departments 테이블에 있는 모든 자료를 기준으로 사원번호, 이름, 업무코드, 부서번호(employees 테이블), 부서번호(departments 테이블), 부서명 조회

```
SELECT emp.employee_id, emp. first_name, emp.job_id, emp.department_id,
dept.department_id, dept.department_name
FROM employees emp, departments dept
WHERE emp.department_id(+)=dept.department_id;
```

기준테이블:departments 테이블

Run SQL Command Line

```
SQL> SELECT emp.employee_id, emp. first_name, emp.job_id, emp.department_id, dept.department_id, dept.department_name
  2  FROM employees emp, departments dept
  3  WHERE dept.department_id=emp.department_id(+);

EMPLOYEE_ID FIRST_NAME                               JOB_ID          DEPARTMENT_ID DEPARTMENT_ID DEPARTMENT_NAME
----------- ---------------------------------------- --------------- ------------- ------------- ------------------------------
        200 Jennifer                                 AD_ASST                    10            10 Administration
        210 Mickey                                   PR_REP                     10            10 Administration
        201 Michael                                  MK_MAN                     20            20 Marketing
        209 M%ary                                    FI_MGR                     20            20 Marketing
        202 Pat                                      MK_REP                     20            20 Marketing
        119 Karen                                    PU_CLERK                   30            30 Purchasing
        118 Guy                                      PU_CLERK                   30            30 Purchasing
        116 Shelli                                   PU_CLERK                   30            30 Purchasing
        115 Alexander                                PU_CLERK                   30            30 Purchasing
        114 Den                                      PU_MAN                     30            30 Purchasing
        208 An%%dy                                   FI_ACCOUNT                 30            30 Purchasing

EMPLOYEE_ID FIRST_NAME                               JOB_ID          DEPARTMENT_ID DEPARTMENT_ID DEPARTMENT_NAME
----------- ---------------------------------------- --------------- ------------- ------------- ------------------------------
        117 Sigal                                    PU_CLERK                   30            30 Purchasing
        203 Susan                                    HR_REP                     40            40 Human Resources
                                                                                              190 Contracting
                                                                                              200 Operations
                                                                                              210 IT Support
                                                                                              220 NOC

EMPLOYEE_ID FIRST_NAME                               JOB_ID          DEPARTMENT_ID DEPARTMENT_ID DEPARTMENT_NAME
----------- ---------------------------------------- --------------- ------------- ------------- ------------------------------
                                                                                              230 IT Helpdesk
                                                                                              240 Government Sales
                                                                                              250 Retail Sales
                                                                                              260 Recruiting
                                                                                              270 Payroll

126 rows selected.

SQL>
```

165	David	SA_REP	80	80	Sales
166	Sundar	SA_REP	80	80	Sales
167	Amit	SA_REP	80	80	Sales
168	Lisa	SA_REP	80	80	Sales
169	Harrison	SA_REP	80	80	Sales
175	Alyssa	SA_REP	80	80	Sales
176	Jonathon	SA_REP	80	80	Sales
177	Jack	SA_REP	80	80	Sales
179	Charles	SA_REP	80	80	Sales
170	Tayler	SA_REP	80	80	Sales
172	Elizabeth	SA_REP	80	80	Sales
171	William	SA_REP	80	80	Sales
102	Lex	AD_VP	90	90	Executive
100	Steven	AD_PRES	90	90	Executive
101	Neena	AD_VP	90	90	Executive
112	Jose Manuel	FI_ACCOUNT	100	100	Finance
111	Ismael	FI_ACCOUNT	100	100	Finance
110	John	FI_ACCOUNT	100	100	Finance
109	Daniel	FI_ACCOUNT	100	100	Finance
113	Luis	FI_ACCOUNT	100	100	Finance
108	Nancy	FI_MGR	100	100	Finance
206	William	AC_ACCOUNT	110	110	Accounting
205	Shelley	AC_MGR	110	110	Accounting
(null)	(null)	(null)	(null)	120	Treasury
(null)	(null)	(null)	(null)	130	Corporate
(null)	(null)	(null)	(null)	140	Control A
(null)	(null)	(null)	(null)	150	Shareholde
(null)	(null)	(null)	(null)	160	Benefits
(null)	(null)	(null)	(null)	170	Manufactu
(null)	(null)	(null)	(null)	180	Construct
(null)	(null)	(null)	(null)	190	Contracti
(null)	(null)	(null)	(null)	200	Operation
(null)	(null)	(null)	(null)	210	IT Suppor
(null)	(null)	(null)	(null)	220	NOC
(null)	(null)	(null)	(null)	230	IT Helpde
(null)	(null)	(null)	(null)	240	Governmen
(null)	(null)	(null)	(null)	250	Retail Sa
(null)	(null)	(null)	(null)	260	Recruitin
(null)	(null)	(null)	(null)	270	Payroll

126개 데이터 출력

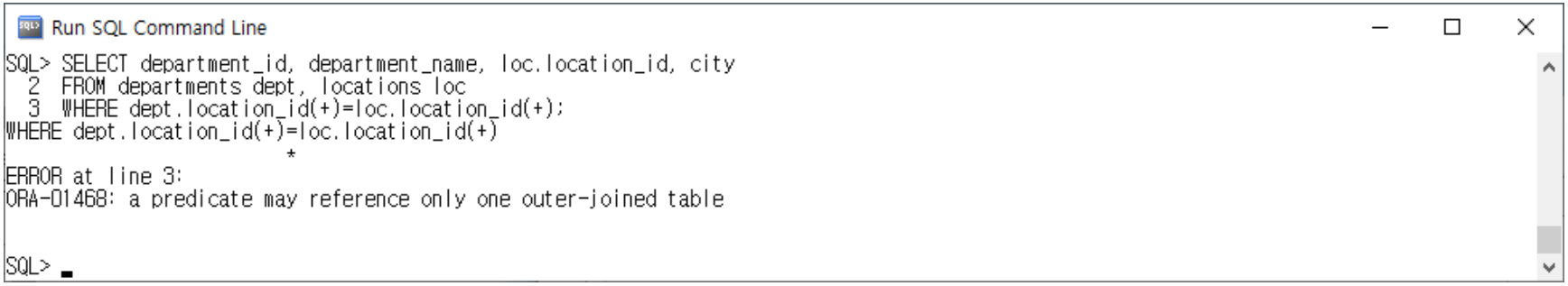
```
SELECT employee_id, first_name, job_id,
emp.department_id, dept.department_id,
department_name
FROM employees emp, departments dept
WHERE
emp.department_id=dept.department_id(+);
```

158	Allan	SA_REP	80	80	Sales
157	Patrick	SA_REP	80	80	Sales
156	Janette	SA_REP	80	80	Sales
155	Oliver	SA_REP	80	80	Sales
154	Nanette	SA_REP	80	80	Sales
153	Christopher	SA_REP	80	80	Sales
152	Peter	SA_REP	80	80	Sales
151	David	SA_REP	80	80	Sales
150	Peter	SA_REP	80	80	Sales
149	Eleni	SA_MAN	80	80	Sales
148	Gerald	SA_MAN	80	80	Sales
147	Alberto	SA_MAN	80	80	Sales
146	Karen	SA_MAN	80	80	Sales
145	John	SA_MAN	80	80	Sales
102	Lex	AD_VP	90	90	Executive
101	Neena	AD_VP	90	90	Executive
100	Steven	AD_PRES	90	90	Executive
113	Luis	FI_ACCOUNT	100	100	Finance
112	Jose Manuel	FI_ACCOUNT	100	100	Finance
111	Ismael	FI_ACCOUNT	100	100	Finance
110	John	FI_ACCOUNT	100	100	Finance
109	Daniel	FI_ACCOUNT	100	100	Finance
108	Nancy	FI_MGR	100	100	Finance
206	William	AC_ACCOUNT	110	110	Accounting
205	Shelley	AC_MGR	110	110	Accounting
178	Kimberely	SA_REP	(null)	(null)	(null)

111개 데이터 출력

다음 결과 확인

```
SELECT department_id, department_name, loc.location_id, city
FROM departments dept, locations loc
WHERE dept.location_id(+)=loc.location_id(+);
```



Syntax Error 발생
발생이유 : outer join table은 하나만 가능
해결방법 : select 문 2개로 UNION 사용

다음 결과 확인(ppt 23장 비교)

```
SELECT employee_id, first_name, job_id,
emp.department_id, dept.department_id,
department_name
FROM employees emp, departments dept
WHERE
emp.department_id=dept.department_id(+)
UNION
SELECT employee_id, first_name, job_id,
emp.department_id, dept.department_id,
department_name
FROM employees emp, departments dept
WHERE
emp.department_id(+)=dept.department_id;
```

127개 데이터 출력

EMPLOYEE_ID	FIRST_NAME	JOB_ID	DEPARTMENT_ID	DEPARTMENT_ID_1	DEPARTMENT_NAME
100	Steven	AD_PRES	90	90	Executive
101	Neena	AD_VP	90	90	Executive
102	Lex	AD_VP	90	90	Executive
103	Alexander	IT_PROG	60	60	IT
104	Bruce	IT_PROG	60	60	IT
105	David	IT_PROG	60	60	IT
106	Valli	IT_PROG	60	60	IT
107	Diana	IT_PROG	60	60	IT
108	Nancy	FI_MGR	100	100	Finance
109	Daniel	FI_ACCOUNT	100	100	Finance
110	John	FI_ACCOUNT	100	100	Finance
176	Jonathon	SA_REP	80	80	Sales
177	Jack	SA_REP	80	80	Sales
178	Kimberely	SA_REP	(null)	(null)	(null)
179	Charles	SA_REP	80	80	Sales
180	Winston	SH_CLERK	50	50	Shipping
181	Jean	SH_CLERK	50	50	Shipping
182	Martha	SH_CLERK	50	50	Shipping
208	An%%dy	FI_ACCOUNT	30	30	Purchasing
209	M%ary	FI_MGR	20	20	Marketing
210	Mickey	PR_REP	10	10	Administration
(null)	(null)	(null)	(null)	120	Treasury
(null)	(null)	(null)	(null)	130	Corporate Tax
(null)	(null)	(null)	(null)	140	Control And Credit
(null)	(null)	(null)	(null)	150	Shareholder Services
(null)	(null)	(null)	(null)	160	Benefits
(null)	(null)	(null)	(null)	170	Manufacturing
(null)	(null)	(null)	(null)	180	Construction
(null)	(null)	(null)	(null)	190	Contracting
(null)	(null)	(null)	(null)	200	Operations
(null)	(null)	(null)	(null)	210	IT Support
(null)	(null)	(null)	(null)	220	NOC
(null)	(null)	(null)	(null)	230	IT Helpdesk
(null)	(null)	(null)	(null)	240	Government Sales
(null)	(null)	(null)	(null)	250	Retail Sales
(null)	(null)	(null)	(null)	260	Recruiting
(null)	(null)	(null)	(null)	270	Payroll

SELF JOIN

- FROM 절에 동일한 테이블을 두 번 이상 사용하여 한 테이블의 행들을 같은 테이블의 행들과 조인
- 한 테이블을 FROM절에 두 번 이상 명시하되, 각각의 테이블을 두 개 이상으로 구분하여 사용하려면 테이블 별칭을 사용하여야 함
- 테이블 하나를 두 개 또는 그 이상으로 SELF JOIN 가능

SELF JOIN 예

- Ellen의 Manager의 이름을 찾는 방법
  - 사원 테이블에서 first_name이 Ellen인 데이터 검색
  - 사원 테이블에서 Ellen의 manager_id 검색
  - 관리자 테이블에서 사원테이블의 manager_id에 해당되는 employee_id 검색
  - 관리자 테이블에서 employee_id에 해당하는 first_name 조회

Run SQL Command Line 사원테이블

EMPLOYEE_ID	FIRST_NAME	JOB_ID	MANAGER_ID
166	Sundar	SA_REP	147
167	Amit	SA_REP	147
168	Lisa	SA_REP	148
169	Harrison	SA_REP	148
170	Tayler	SA_REP	148
171	William	SA_REP	148
172	Elizabeth	SA_REP	148
173	Sundita	SA_REP	148
174	Ellen	SA_REP	149
175	Alyssa	SA_REP	149

Run SQL Command Line 관리자테이블

EMPLOYEE_ID	FIRST_NAME	JOB_ID	MANAGER_ID
144	Peter	ST_CLERK	124
145	John	SA_MAN	100
146	Karen	SA_MAN	100
147	Alberto	SA_MAN	100
148	Gerald	SA_MAN	100
149	Eleni	SA_MAN	100
150	Peter	SA_REP	145
151	David	SA_REP	145
152	Peter	SA_REP	145
153	Christopher	SA_REP	145

```
SELECT emp.employee_id 사원번호, emp.first_name 사원명, emp.manager_id 관리자번호,
man.first_name 관리자명
FROM employees emp, employees man
WHERE emp.first_name='Ellen' AND emp.manager_id=man.employee_id;
```

사원번호	사원명	관리자번호	관리자명
174	Ellen	149	Eleni

SQL>

ANSI JOIN

- **FROM 절에서 바로 JOIN을 명시적으로 정의**

```
SELECT table1.column, table2.column, ...
FROM table1
  [CROSS JOIN table2] |
  [NATURAL JOIN table2] |
  [JOIN table2 USING (column_name)] |
  [JOIN table2 ON (table1.column_name = table2.column_name)] |
  [LEFT|RIGHT|FULL OUTER JOIN table2 ON (table1.column_name
= table2.column_name)];
```

ANSI JOIN 예시1

```
SELECT * FROM employees NATURAL JOIN departments;
```

Run SQL Command Line

```
SQL> SELECT * FROM employees NATURAL JOIN departments;
```

MANAGER_ID	DEPARTMENT_ID	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	DEPARTMENT_NAME	LOCATION_ID
100	90	101	Neena	Kochhar	NKOCHHAR	515.123.4568	05/09/21	AD_VP	17000		Executive	1700
100	90	102	Lex	De Haan	LDEHAAN	515.123.4569	01/01/13	AD_VP	17000		Executive	1700
103	60	104	Bruce	Ernst	BERNST	590.423.4568	07/05/21	IT_PROG	6000		IT	1400
103	60	105	David	Austin	DAUSTIN	590.423.4569	05/06/25	IT_PROG	4800		IT	1400
103	60	106	Valli	Pataballa	VPATABAL	590.423.4560	06/02/05	IT_PROG	4800		IT	1400
103	60	107	Diana	Lorentz	DLORENTZ	590.423.5567	07/02/07	IT_PROG	4200		IT	1400
108	100	109	Daniel	Faviet	DFAVIET	515.124.4169	02/08/16	FI_ACCOUNT	9000		Finance	1700
108	100	110	John	Chen	JCHEN	515.124.4269	05/09/28	FI_ACCOUNT	8200		Finance	1700
108	100	111	Ismael	Sciarra	ISCIARRA	515.124.4369	05/09/30	FI_ACCOUNT	7700		Finance	1700
108	100	112	Jose Manuel	Urman	JMURMAN	515.124.4469	06/03/07	FI_ACCOUNT	7800		Finance	1700
108	100	113	Luis	Popp	LPOPP	515.124.4567	07/12/07	FI_ACCOUNT	6900		Finance	1700

MANAGER_ID	DEPARTMENT_ID	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	DEPARTMENT_NAME	LOCATION_ID
114	30	115	Alexander	Khoo	AKHOO	515.127.4562	03/05/18	PU_CLERK	3100		Purchasing	1700
114	30	116	Shelli	Baida	SBAIDA	515.127.4563	05/12/24	PU_CLERK	2900		Purchasing	1700
114	30	117	Sigal	Tobias	STOBIAS	515.127.4564	05/07/24	PU_CLERK	2800		Purchasing	1700
114	30	118	Guy	Himuro	GHIMURO	515.127.4565	06/11/15	PU_CLERK	2600		Purchasing	1700
114	30	119	Karen	Colmenares	KCOLMENA	515.127.4566	07/08/10	PU_CLERK	2500		Purchasing	1700
121	50	120	Laura	Bissot	LBISSOT	650.124.5234	05/08/20	ST_CLERK	3300		Shipping	1500
121	50	130	Mozhe	Atkinson	MATKINSO	650.124.6234	05/10/30	ST_CLERK	2800		Shipping	1500
121	50	131	James	Marlow	JAMRLOW	650.124.7234	05/02/16	ST_CLERK	2500		Shipping	1500
121	50	132	TJ	Olson	TJOLSON	650.124.8234	07/04/10	ST_CLERK	2100		Shipping	1500
145	80	150	Peter	Tucker	PTUCKER	011.44.1344.129268	05/01/30	SA_REP	10000	.3	Sales	2500
145	80	151	David	Bernstein	DBERNSTE	011.44.1344.345268	05/03/24	SA_REP	9500	.25	Sales	2500

MANAGER_ID	DEPARTMENT_ID	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	DEPARTMENT_NAME	LOCATION_ID
145	80	152	Peter	Hall	PHALL	011.44.1344.478968	05/08/20	SA_REP	9000	.25	Sales	2500
145	80	153	Christopher	Olsen	COLSEN	011.44.1344.498718	06/03/30	SA_REP	8000	.2	Sales	2500
145	80	154	Nanette	Cambrault	NCAMBRAU	011.44.1344.987668	06/12/09	SA_REP	7500	.2	Sales	2500
145	80	155	Oliver	Tuvault	OTUVAULT	011.44.1344.486508	07/11/23	SA_REP	7000	.15	Sales	2500
121	50	164	Nandita	Sarchand	NSARCHAN	650.509.1876	04/01/27	SH_CLERK	4200		Shipping	1500
121	50	165	Alexis	Bull	ABULL	650.509.2876	05/02/20	SH_CLERK	4100		Shipping	1500
121	50	166	Julia	Dellinger	JDELLING	650.509.3876	06/06/24	SH_CLERK	3400		Shipping	1500
121	50	167	Anthony	Cabrio	ACABRIO	650.509.4876	07/02/07	SH_CLERK	3000		Shipping	1500
201	20	202	Pat	Fay	PFAY	603.123.6666	05/08/17	MK_REP	6000		Marketing	1800
205	110	206	William	Gietz	WGIETZ	515.123.8181	02/06/07	AC_ACCOUNT	8300		Accounting	1700

```
80 rows selected.
SQL>
```

```
SELECT * FROM employees emp, departments dept
WHERE emp.department_id=dept.department_id AND emp.manager_id=dept.manager_id;
```

ANSI JOIN 예시2

```
SELECT * FROM employees JOIN departments USING (department_id);
```

Run SQL Command Line

```
SQL> SELECT * FROM employees JOIN departments using (department_id);
```

DEPARTMENT_ID	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
10	200	Jennifer	Whalen	JWHALEN	515.123.4444	03/09/17	AD_ASST	4400		101	Administration	200	1700
10	210	Mickey	Mouse	MouseMAIL	987.654.3210	03/08/25	PR_REP		0	124	Administration	200	1700
20	201	Michael	Hartstein	MHARTSTE	515.123.5555	04/02/17	MK_MAN	13000		100	Marketing	201	1800
20	209	Mary	Queen	QueenMAIL	123.456.7890	11/11/14	FI_MGR	9400		114	Marketing	201	1800
20	202	Pat	Fay	PFAY	603.123.6666	05/08/17	MK_REP	6000		201	Marketing	201	1800
30	119	Karen	Colmenares	KCOLMENA	515.127.4566	07/08/10	PU_CLERK	2500		114	Purchasing	114	1700
30	118	Guy	Himuro	GHIMURO	515.127.4565	06/11/15	PU_CLERK	2600		114	Purchasing	114	1700
30	116	Shelli	Baida	SBAIDA	515.127.4563	05/12/24	PU_CLERK	2900		114	Purchasing	114	1700
30	115	Alexander	Khoo	AKHOO	515.127.4562	03/05/18	PU_CLERK	3100		114	Purchasing	114	1700
30	114	Den	Raphaely	DRAPHEAL	515.127.4561	02/12/07	PU_MAN	11000		100	Purchasing	114	1700
30	206	Antkdy	Chaplin	ChaplinMAIL	515.195.9876	12/09/27	FI_ACCOUNT	2400		100	Purchasing	114	1700

DEPARTMENT_ID	EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
30	117	Sigal	Tobias	STOBIAS	515.127.4564	05/07/24	PU_CLERK	2800		114	Purchasing	114	1700
40	203	Susan	Mavris	SMAVRIS	515.123.7777	02/06/07	HR_REP	6500		101	Human Resources	203	2400
50	120	Matthew	Weiss	MWEISS	650.123.1234	04/07/18	ST_MAN	8000		100	Shipping	121	1500
50	121	Adam	Fripp	AFRIPP	650.123.2234	05/04/10	ST_MAN	8200		100	Shipping	121	1500
50	122	Payam	Kaufling	PKAUFLIN	650.123.3234	03/05/01	ST_MAN	7900		100	Shipping	121	1500
50	123	Shanta	Vollman	SVOLLMAN	650.123.4234	05/10/10	ST_MAN	6500		100	Shipping	121	1500
50	124	Kevin	Mourgos	KMOURGOS	650.123.5234	07/11/16	ST_MAN	5800		100	Shipping	121	1500
50	125	Julia	Nayer	JNAYER	650.124.1214	05/07/16	ST_CLERK	3200		120	Shipping	121	1500
50	126	Irene	Mikkilineni	IMIKKILI	650.124.1224	06/09/28	ST_CLERK	2700		120	Shipping	121	1500
90	100	Steven	King	SKING	515.123.4567	03/06/17	AD_PRES	24000			Executive	100	1700
90	101	Neena	Kochhar	NKOCHHAR	515.123.4568	05/09/21	AD_VP	17000		100	Executive	100	1700
100	112	Jose Manuel	Urman	JMURMAN	515.124.4469	06/03/07	FI_MGR	7800		101	Finance	108	1700
100	111	Ismael	Sciarra	ISCIARRA	515.124.4369	05/09/30	FI_ACCOUNT	7700		108	Finance	108	1700
100	110	John	Chen	JCHEN	515.124.4269	05/09/28	FI_ACCOUNT	8200		108	Finance	108	1700
100	109	Daniel	Faviet	DFAVIET	515.124.4169	02/08/16	FI_ACCOUNT	9000		108	Finance	108	1700
100	113	Luis	Popp	LPOPP	515.124.4567	07/12/07	FI_ACCOUNT	6900		108	Finance	108	1700
100	108	Nancy	Greenberg	NGREENBE	515.124.4569	02/08/17	FI_MGR	12008		101	Finance	108	1700
110	206	William	Gietz	WGIETZ	515.123.8181	02/06/07	AC_ACCOUNT	8300		205	Accounting	205	1700
110	205	Shelley	Higgins	SHIGGINS	515.123.8080	02/06/07	AC_MGR	12008		101	Accounting	205	1700

```
110 rows selected.

SQL>
```

```
SELECT * FROM employees JOIN departments ON
employees.department_id=departments.department_id;
```

```
SELECT * FROM employees emp, departments dept
WHERE emp.department_id=dept.department_id;
```

ANSI JOIN 예시3

```
SELECT * FROM employees RIGHT OUTER JOIN departments ON
employees.department_id=departments.department_id;
```

Run SQL Command Line

```
SQL> SELECT * FROM employees RIGHT OUTER JOIN departments ON employees.department_id=departments.department_id;
```

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
200	Jennifer	Whalen	JWHALEN	515.123.4444	03/09/17	AD_ASST	4400		101	10	10	Administration	200	1700
210	Mickey	Mouse	MouseMAIL	987.654.3210	03/06/25	PR_REP	0		124	10	10	Administration	200	1700
201	Michael	Hartstein	MHARTSTE	515.123.5555	04/02/17	MK_MAN	13000		100	20	20	Marketing	201	1800
209	MSary	Queen	QueenMAIL	123.456.7890	11/11/14	FI_MGR	8400		114	20	20	Marketing	201	1800
202	Pat	Fay	PFAY	603.123.6666	05/08/17	MK_REP	6000		201	20	20	Marketing	201	1800
119	Karen	Colmenares	KCOLMENA	515.127.4566	07/08/10	PU_CLERK	2500		114	30	30	Purchasing	114	1700
118	Guy	Himuro	GHIMURO	515.127.4565	06/11/15	PU_CLERK	2600		114	30	30	Purchasing	114	1700
116	Shelli	Baida	SBAIDA	515.127.4563	05/12/24	PU_CLERK	2900		114	30	30	Purchasing	114	1700
115	Alexander	Khoo	AKHOO	515.127.4562	03/05/18	PU_CLERK	3100		114	30	30	Purchasing	114	1700
114	Den	Raphaely	DRAPHEAL	515.127.4561	02/12/07	PU_MAN	11000		100	30	30	Purchasing	114	1700
206	AnNdy	Chaplin	ChaplinMAIL	515.135.9876	12/03/27	FI_ACCOUNT	2400		100	30	30	Purchasing	114	1700

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
117	Sigal	Tobias	STOBIAS	515.127.4564	05/07/24	PU_CLERK	2800		114	30	30	Purchasing	114	1700
203	Susan	Mavris	SMAVRIS	515.123.7777	02/06/07	HR_REP	6500		101	40	40	Human Resources	203	2400
120	Matthew	Weiss	MWEISS	650.123.1234	04/07/18	ST_MAN	8000		100	50	50	Shipping	121	1500
121	Adam	Fripp	AFRIPP	650.123.2234	05/04/10	ST_MAN	8200		100	50	50	Shipping	121	1500
122	Payam	Kaufling	PKAUFLIN	650.123.3234	03/05/01	ST_MAN	7900		100	50	50	Shipping	121	1500
123	Shanta	Vollman	SVOLLMAN	650.123.4234	05/10/10	ST_MAN	6500		100	50	50	Shipping	121	1500
124	Kevin	Mourgos	KMOURGOS	650.123.5234	07/11/16	ST_MAN	5800		100	50	50	Shipping	121	1500
125	Julia	Nayer	JNAYER	650.124.1214	05/07/16	ST_CLERK	3200		120	50	50	Shipping	121	1500
126	Irene	Mikkilineni	IMIKKILI	650.124.1224	06/09/28	ST_CLERK	2700		120	50	50	Shipping	121	1500
127	James	Landry	JLANDRY	650.124.1334	07/01/14	ST_CLERK	2400		120	50	50	Shipping	121	1500
128	Steven	Markle	SMARKLE	650.124.1434	08/03/08	ST_CLERK	2200		120	50	50	Shipping	121	1500

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
129	Laura	Bissot	LBISSOT	650.124.5234	05/08/20	ST_CLERK	3300		121	50	50	Shipping	121	1500
130	Mozhe	Atkinson	MATKINSO	650.124.6234	05/10/30	ST_CLERK	2800		121	50	50	Shipping	121	1500
131	James	Marlow	JAMRLOW	650.124.7234	05/02/16	ST_CLERK	2500		121	50	50	Shipping	121	1500
132	TJ	Olson	TJOLSON	650.124.8234	07/04/10	ST_CLERK	2100		121	50	50	Shipping	121	1500
133	Jason	Mallin	JMALLIN	650.127.1934	04/06/14	ST_CLERK	3300		122	50	50	Shipping	121	1500
134	Michael	Rogers	MROGERS	650.127.1834	06/08/26	ST_CLERK	2900		122	50	50	Shipping	121	1500
135	Ki	Gee	KGEE	650.127.1734	07/12/12	ST_CLERK	2400		122	50	50	Shipping	121	1500
136	Hazel	Philtanker	HPHILTAN	650.127.1634	08/02/06	ST_CLERK	2200		122	50	50	Shipping	121	1500
137	Renske	Ladwig	RLADWIG	650.121.1234	03/07/14	ST_CLERK	3600		123	50	50	Shipping	121	1500
138	Stephen	Stiles	SSTILES	650.121.2034	05/10/26	ST_CLERK	3200		123	50	50	Shipping	121	1500
139	John	Seo	JSEO	650.121.2019	06/02/12	ST_CLERK	2700		123	50	50	Shipping	121	1500

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
140	Joshua	Patel	JPATEL	650.121.1834	06/04/06	ST_CLERK	2500		123	50	50	Shipping	121	1500
141	Trenna	Rajs	TRAJS	650.121.8009	03/10/17	ST_CLERK	3500		124	50	50	Shipping	121	1500
142	Curtis	Davies	CDAVIES	650.121.2994	05/01/29	ST_CLERK	3100		124	50	50	Shipping	121	1500
143	Randall	Matos	RMATOS	650.121.2874	06/03/15	ST_CLERK	2600		124	50	50	Shipping	121	1500
144	Peter	Vargas	PVARGAS	650.121.2004	06/07/09	ST_CLERK	2500		124	50	50	Shipping	121	1500
180	Winston	Taylor	WTAYLOR	650.507.9876	06/01/24	SH_CLERK	3200		120	50	50	Shipping	121	1500
181	Jean	Fleaur	JFLEAUR	650.507.9877	06/02/23	SH_CLERK	3100		120	50	50	Shipping	121	1500
182	Martha	Sullivan	MSULLIVA	650.507.9878	07/06/21	SH_CLERK	2500		120	50	50	Shipping	121	1500
183	Girard	Geoni	GGEONI	650.507.9879	08/02/03	SH_CLERK	2800		120	50	50	Shipping	121	1500
108	Nancy	Greenberg	NGREENBE	515.124.4569	02/08/17	FI_MGR	12008		101	100	100	Finance	108	1700
206	William	Gietz	WGIETZ	515.123.8181	02/06/07	AC_ACCOUNT	8300		205	110	110	Accounting	205	1700
205	Shelley	Higgins	SHIGGINS	515.123.8080	02/06/07	AC_MGR	12008		101	110	110	Accounting	205	1700

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
											120	Treasury		1700
											130	Corporate Tax		1700
											140	Control And Credit		1700
											150	Shareholder Services		1700
											160	Benefits		1700
											170	Manufacturing		1700
											180	Construction		1700
											190	Contracting		1700
											200	Operations		1700
											210	IT Support		1700
											220	NOC		1700

EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DAT	JOB_ID	SALARY	COMMISSION_PCT	MANAGER_ID	DEPARTMENT_ID	DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
											230	IT Helpdesk		1700
											240	Government Sales		1700
											250	Retail Sales		1700
											260	Recruiting		1700
											270	Payroll		1700

```
126 rows selected.
SQL>
```

```
SELECT * FROM employees LEFT OUTER JOIN departments ON
employees.department_id=departments.department_id;
```

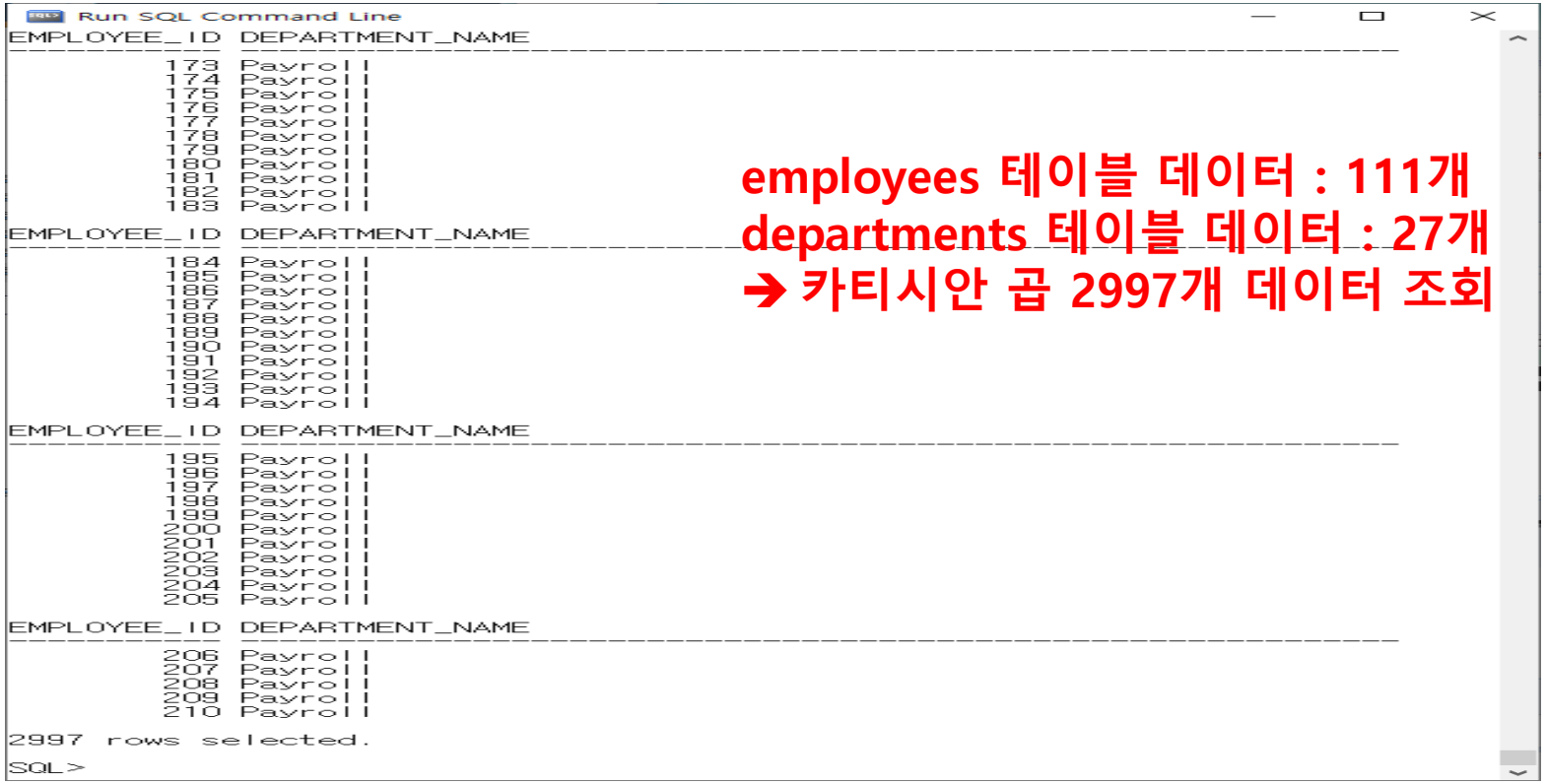
Kimberly확인

CROSS JOIN

- 두 테이블 상호간의 조합 생성
- 두 테이블 사이의 카티시안곱(Cartesian Product)과 동일

CROSS JOIN 예시

```
SELECT emp.employee_id, dept.department_name
FROM employees emp CROSS JOIN departments dept;
```



NATURAL JOIN

- 두 테이블에서 동일한 이름을 가진 모든 열을 기준으로 조인
- 두 테이블의 일치하는 모든 열에서 같은 값을 가진 행을 선택
- 조인조건으로 사용한 컬럼 앞에는 테이블명이나 테이블 별칭을 명시할 수 없다.
- WHERE 절을 사용하여 조건 추가 가능

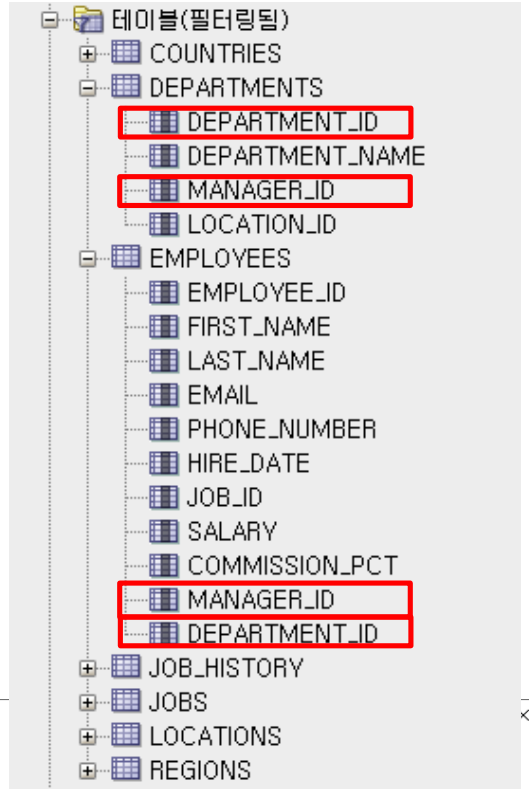
```
Run SQL Command Line
SQL> SELECT * FROM jobs;

JOB_ID               JOB_TITLE                                                          MIN_SALARY MAX_SALARY
-------------------- ------------------------------------------------------------------ ---------- ----------
AD_PRES              President                                                               20080      40000
AD_VP                Administration Vice President                                           15000      30000
AD_ASST              Administration Assistant                                                 3000       6000
FI_MGR               Finance Manager                                                          8200      16000
FI_ACCOUNT           Accountant                                                               4200       9000
AC_MGR               Accounting Manager                                                       8200      16000
AC_ACCOUNT           Public Accountant                                                        4200       9000
SA_MAN               Sales Manager                                                           10000      20080
SA_REP               Sales Representative                                                     6000      12008
PU_MAN               Purchasing Manager                                                       8000      15000
PU_CLERK             Purchasing Clerk                                                         2500       5500

JOB_ID               JOB_TITLE                                                          MIN_SALARY MAX_SALARY
-------------------- ------------------------------------------------------------------ ---------- ----------
ST_MAN               Stock Manager                                                            5500       8500
ST_CLERK             Stock Clerk                                                              2008       5000
SH_CLERK             Shipping Clerk                                                           2500       5500
IT_PROG              Programmer                                                               4000      10000
MK_MAN               Marketing Manager                                                        9000      15000
MK_REP               Marketing Representative                                                 4000       9000
HR_REP               Human Resources Representative                                           4000       9000
PR_REP               Public Relations Representative                                          4500      10500

19 rows selected.

SQL> SELECT * FROM job_history;

EMPLOYEE_ID START_DA END_DATE JOB_ID              DEPARTMENT_ID
----------- -------- -------- -------------------- -------------
        102 01/01/13 06/07/24 IT_PROG                         60
        101 97/09/21 01/10/27 AC_ACCOUNT                     110
        101 01/10/28 05/03/15 AC_MGR                         110
        201 04/02/17 07/12/19 MK_REP                          20
        114 06/03/24 07/12/31 ST_CLERK                        50
        122 07/01/01 07/12/31 ST_CLERK                        50
        200 95/09/17 01/06/17 AD_ASST                         90
        176 06/03/24 06/12/31 SA_REP                          80
        176 07/01/01 07/12/31 SA_MAN                          80
        200 02/07/01 06/12/31 AC_ACCOUNT                      90

10 rows selected.

SQL> _
```

```
Run SQL Command Line
SQL> SELECT * FROM jobs NATURAL JOIN job_history;

JOB_ID               JOB_TITLE                                                          MIN_SALARY MAX_SALARY EMPLOYEE_ID START_DA END_DATE DEPARTMENT_ID
-------------------- ------------------------------------------------------------------ ---------- ---------- ----------- -------- -------- -------------
AC_ACCOUNT           Public Accountant                                                        4200       9000         101 97/09/21 01/10/27           110
AC_ACCOUNT           Public Accountant                                                        4200       9000         200 02/07/01 06/12/31            90
AC_MGR               Accounting Manager                                                       8200      16000         101 01/10/28 05/03/15           110
AD_ASST              Administration Assistant                                                 3000       6000         200 95/09/17 01/06/17            90
IT_PROG              Programmer                                                               4000      10000         102 01/01/13 06/07/24            60
MK_REP               Marketing Representative                                                 4000       9000         201 04/02/17 07/12/19            20
SA_MAN               Sales Manager                                                           10000      20080         176 07/01/01 07/12/31            80
SA_REP               Sales Representative                                                     6000      12008         176 06/03/24 06/12/31            80
ST_CLERK             Stock Clerk                                                              2008       5000         122 07/01/01 07/12/31            50
ST_CLERK             Stock Clerk                                                              2008       5000         114 06/03/24 07/12/31            50

10 rows selected.

SQL> _
```

Run SQL Command Line

```
SQL> SELECT first_name, department_id, manager_id  FROM employees;

FIRST_NAME                               DEPARTMENT_ID MANAGER_ID
---------------------------------------- ------------- ----------
An%%dy                                              30        100
M%ary                                               20        114
Mickey                                              10        124
Steven                                              90
Neena                                               90        100
Lex                                                 90        100
Alexander                                           60        102
Bruce                                               60        103
David                                               60        103
Valli                                               60        103
Diana                                               60        103

FIRST_NAME                               DEPARTMENT_ID MANAGER_ID
---------------------------------------- ------------- ----------
Nancy                                              100        101
Daniel                                             100        108
John                                               100        108
Ismael                                             100        108
Jose Manuel                                        100        108
Luis                                               100        108
Den                                                 30        100
Alexander                                           30        114
Shelli                                              30        114
Sigal                                               30        114
Guy                                                 30        114

FIRST_NAME                               DEPARTMENT_ID MANAGER_ID
---------------------------------------- ------------- ----------
Karen                                               30        114
Matthew                                             50        100
Adam                                                50        100
Payam                                               50        100
Shanta                                              50        100
Kevin                                               50        100
Julia                                               50        120
Irene                                               50        120
James                                               50        120
Steven                                              50        120
Laura                                               50        121

FIRST_NAME                               DEPARTMENT_ID MANAGER_ID
---------------------------------------- ------------- ----------
Mozhe                                               50        121
James                                               50        121
TJ                                                  50        121
Jason                                               50        122
Michael                                             50        122
Ki                                                  50        122
Hazel                                               50        122
Renske                                              50        123
Stephen                                             50        123
John                                                50        123
Joshua                                              50        123

FIRST_NAME                               DEPARTMENT_ID MANAGER_ID
---------------------------------------- ------------- ----------
Trenna                                              50        124
Curtis                                              50        124
Randall                                             50        124
Peter                                               50        124
John                                                80        100
Karen                                               80        100
Alberto                                             80        100
```

설명



Run SQL Command Line

```
SQL> SELECT * FROM departments;

DEPARTMENT_ID DEPARTMENT_NAME                                  MANAGER_ID LOCATION_ID
------------- ---------------------------------------------- ---------- -----------
           10 Administration                                        200        1700
           20 Marketing                                             201        1800
           30 Purchasing                                            114        1700
           40 Human Resources                                       203        2400
           50 Shipping                                              121        1500
           60 IT                                                    103        1400
           70 Public Relations                                      204        2700
           80 Sales                                                 145        2500
           90 Executive                                             100        1700
          100 Finance                                               108        1700
          110 Accounting                                            205        1700

DEPARTMENT_ID DEPARTMENT_NAME                                  MANAGER_ID LOCATION_ID
------------- ---------------------------------------------- ---------- -----------
          120 Treasury                                                         1700
          130 Corporate Tax                                                    1700
          140 Control And Credit                                               1700
          150 Shareholder Services                                             1700
          160 Benefits                                                         1700
          170 Manufacturing                                                    1700
          180 Construction                                                     1700
          190 Contracting                                                      1700
          200 Operations                                                       1700
          210 IT Support                                                       1700
          220 NOC                                                              1700

DEPARTMENT_ID DEPARTMENT_NAME                                  MANAGER_ID LOCATION_ID
------------- ---------------------------------------------- ---------- -----------
          230 IT Helpdesk                                                      1700
          240 Government Sales                                                 1700
          250 Retail Sales                                                     1700
          260 Recruiting                                                       1700
          270 Payroll                                                          1700

27 rows selected.
```

NATURAL JOIN 테스트

TABLE 생성(test_join, test_natural_join) 데이터 삽입 NATURAL JOIN 테스트

데이터 삽입 NATURAL JOIN 테스트

Run SQL Command Line

```
SQL> CREATE TABLE test_join(
  2     test varchar2(20),
  3     no number(2)
  4  );

Table created.

SQL> CREATE TABLE test_natural_join(
  2     job_id varchar2(10),
  3     job_title varchar(25) NOT NULL,
  4     test number(10)
  5  );

Table created.
```

Run SQL Command Line

```
SQL> select * from tab;

TNAME                                              TABTYPE        CLUSTERID
-------------------------------------------------- ------------- ----------
COUNTRIES                                          TABLE
DEPARTMENTS                                        TABLE
EMPLOYEES                                          TABLE
EMP_DETAILS_VIEW                                   VIEW
JOBS                                               TABLE
JOB_HISTORY                                        TABLE
LOCATIONS                                          TABLE
REGIONS                                            TABLE
TEST_JOIN                                          TABLE
TEST_NATURAL_JOIN                                  TABLE

10 rows selected.
```

Run SQL Command Line

```
SQL> SELECT * FROM test_join NATURAL JOIN test_natural_join;

      TEST         NO JOB_ID                JOB_TITLE
---------- ---------- --------------------- -------------------------
      1000         20 10                    TEST_JOB
```

Run SQL Command Line

```
SQL> INSERT INTO test_join  VALUES('1000', 20);

1 row created.

SQL> INSERT INTO test_natural_join  VALUES('10', 'TEST_JOB', 1000);

1 row created.
```

Run SQL Command Line

```
SQL> INSERT INTO test_join  VALUES('ABC', 20);

1 row created.
```

Run SQL Command Line

```
SQL> SELECT * FROM test_join NATURAL JOIN test_natural_join;
SELECT * FROM test_join NATURAL JOIN test_natural_join
*
ERROR at line 1:
ORA-01722: invalid number


SQL>
```

JOIN ~ USING

- NATURAL JOIN은 이름과 데이터 유형이 일치하는 모든 열을 사용하여 테이블을 조인하지만 USING 절을 사용하면 등가(=) 조인의 특정 열 지정 가능

- USING 절에 참조되는 열은 SQL 문 어디에서도 테이블 명이나 별칭을 가질 수 없다.

- NATURAL JOIN 과 USING 절은 서로 배타적으로 사용

JOIN ~ ON

- NATURAL JOIN의 조건은 기본적으로 같은 이름을 가진 모든 열의 등가(=) 조인
- 임의의 열을 지정하거나 조인할 열을 지정하려면 ON 절 사용
- 조인 조건이 다른 검색조건과 분리
- ON 절을 사용하면 코드 이해도가 높아짐

OUTER JOIN

- ANSI JOIN에서 INNER JOIN
  - 두 테이블을 조인해서 일치하는 행만 반환하는 조인
- LEFT OUTER JOIN(RIGHT OUTER JOIN)
  - 두 테이블을 조인해서 내부 조인의 결과와 함께 일치하지 않는 왼쪽(오른쪽) 테이블의 행을 반환하는 조인
- FULL OUTER JOIN
  - 두 테이블을 조인해서 내부 조인의 결과와 함께 왼쪽, 오른쪽 조인의 결과를 모두 반환하는 조인