

## 4. 테이블 조인

컴퓨터공학부

김은경

## [학습 목표]

1. 조인(Join)이 무엇인지 정의할 수 있다.
2. 카티션 프로덕트(Cartesian Product)에 대해서 설명할 수 있다.
3. 동일 조인(Equi Join)을 위한 SELECT 문을 작성할 수 있다.
4. 외부 조인(Outer Join)을 위한 SELECT 문을 작성할 수 있다.
5. 자체 조인(Self Join)을 위한 SELECT 문을 작성할 수 있다.

# 1. 조인 개요

## ★ 조인(Join)이란?

- 하나 이상의 테이블로부터 필요한 데이터를 검색하기 위한 일종의 연산

```
SELECT  [테이블이름1.]열이름1, [테이블이름2.]열이름2, ...  
FROM    테이블이름1, 테이블이름2  
WHERE   조인_조건(들);
```

- 테이블 1과 테이블 2를 조인 조건에 따라 조인한 다음, 열 이름1, 열 이름2 등을 표시함

### ✓ 작성 규칙

- ① SELECT 절의 테이블 이름은 생략할 수 있음
- ② 단, 다른 테이블에 같은 열 이름이 사용된 경우, 구분을 위해서 열 이름 앞에 반드시 테이블 이름을 기술하고, (.dot, 점)으로 구분해야 함
- ③ FROM 절에 조인할 모든 테이블 이름을 기술함
- ④ WHERE 절에 조인 조건을 기술함
- ⑤ n개 테이블을 조인할 때는 최소 n-1개의 조인 조건을 기술해야 함

# 1. 조인 개요

## ✓ 테이블 조인 예

- 모든 사원의 이름과 근무부서번호와 근무처, 즉 부서 위치를 검색하기

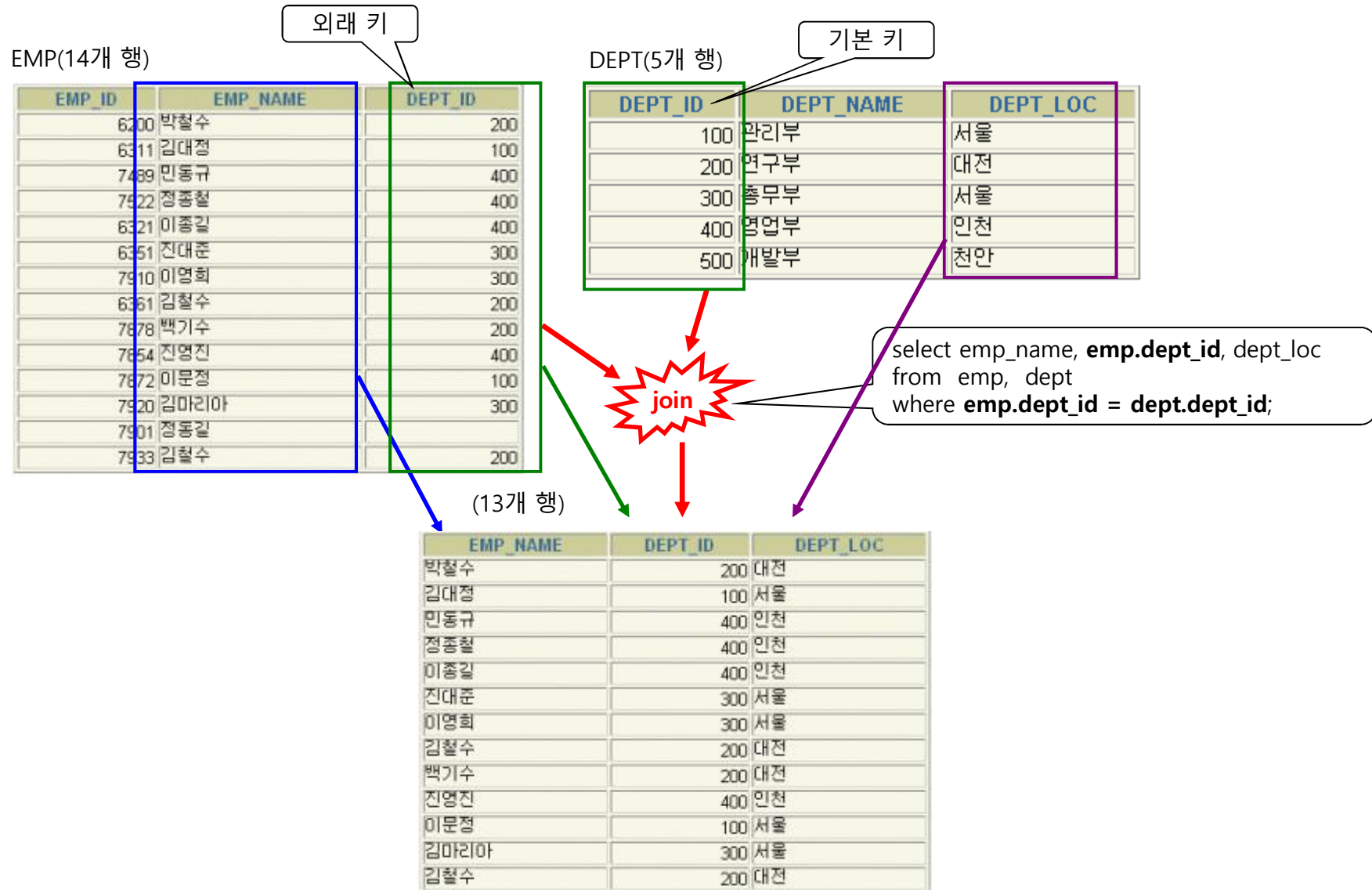
```
SELECT emp_name, emp.dept_id, dept_loc  
FROM emp, dept  
WHERE emp.dept_id = dept.dept_id;
```

EMP_NAME	DEPT_ID	DEPT_LOC
박철수	200	대전
김대정	100	서울
민동규	400	인천
정종철	400	인천
이종길	400	인천
진대준	300	서울
이영희	300	서울
김철수	200	대전
백기수	200	대전
진영진	400	인천
이문정	100	서울
김마리아	300	서울
김철수	200	대전

13 rows selected.

# 1. 조인 개요

## ✓ 참고



# 1. 조인 개요

## ★ 카티션 프로덕트(Cartesian Product)란?

- 첫 번째 테이블의  $m$ 개의 모든 행이 두 번째 테이블의  $n$ 개의 모든 행과 조인되어, 전체  $m*n$ 개의 행이 출력되는 연산

## ★ 카티션 프로덕트가 발생하는 원인

- ① 조인 조건을 생략한 경우
- ② 조인 조건을 잘못 기술한 경우

## ★ 카티션 프로덕트 방지 방안

- $n$ 개 테이블을 조인하는 경우, 최소  $n-1$ 개의 조건을 기술함

# 1. 조인 개요

## ✓ 카티션 프로젝트 사례 : 조인 조건 생략

```
SELECT emp_name, emp.dept_id, dept_loc
FROM emp, dept;
```

EMP(14개 행)

EMP_ID	EMP_NAME	DEPT_ID
6200	박철수	200
6311	김대정	100
7489	민동규	400
7522	정종철	400
6321	이종길	400
6351	진대준	300
7910	이영희	300
6361	김철수	200
7878	백기수	200
7854	진영진	400
7872	이문정	100
7920	김마리아	300
7901	정동길	
7933	김철수	200

DEPT(5개 행)

DEPT_ID	DEPT_NAME	DEPT_LOC
100	관리부	서울
200	연구부	대전
300	총무부	서울
400	영업부	인천
500	개발부	천안



EMP_NAME	DEPT_ID	DEPT_LOC
박철수	200	서울
김대정	100	서울
민동규	400	서울
정종철	400	서울

EMP_NAME	DEPT_ID	DEPT_LOC
김철수	200	서울
박철수	200	대전
김대정	100	대전
민동규	400	대전
정종철	400	대전

EMP_NAME	DEPT_ID	DEPT_LOC
김철수	200	인천
박철수	200	천안
김대정	100	천안
민동규	400	천안
정종철	400	천안

EMP_NAME	DEPT_ID	DEPT_LOC
정동길		천안
김철수	200	천안

70 개의 행이 선택되었습니다.

# 1. 조인 개요

## ★ 조인의 4가지 유형

조인 유형	내용
동일 조인 (Equi join)	양쪽 테이블에 서로 직접 관련되는 속성이 있는 경우, <b>동일("=")</b> 연산자로 조인 조건을 표현한 연산을 의미함
비동일 조인 (Non-equi join)	조인할 테이블의 어떤 열도 직접 관련되어 있지 않는 경우의 조인을 의미함
외부 조인 (Outer join)	조인할 테이블에 조인 조건을 만족하는 행이 없는 경우, " <b>널(Null) 행</b> "을 추가해서 결과 테이블에 포함시키는 연산을 의미함
자체 조인 (Self join)	물리적으로 <b>동일한</b> 테이블을 논리적으로 다른 테이블인 것으로 간주하고 조인하는 것을 의미함



## 2. 동일 조인

### ★ 동일 조인(Equi Join)이란?

- 양쪽 테이블에 서로 직접 관련되는 속성이 있는 경우, 조인 조건을 동일 ('=') 연산자로 표현한 조인

```
SELECT  [테이블이름1.]열이름1, [테이블이름2.]열이름2, ...  
FROM    테이블이름1, 테이블이름2  
WHERE   테이블이름1.열이름3 = 테이블이름1.열이름4 ;
```

### ★ 동일 조인의 특징

- 양쪽 테이블의 서로 직접 관련되는 속성은 각각 한 테이블의 "기본 키"와 다른 테이블의 "외래 키"가 됨

## 2. 동일 조인

### ✓ 동일 조인 예

- 모든 사원의 사원번호와 이름, 부서번호, 부서의 위치 검색하기

```
SELECT emp_id, emp_name, emp.dept_id, dept.dept_id, dept_loc
FROM emp, dept
WHERE emp.dept_id = dept.dept_id;
```

EMP_ID	EMP_NAME	DEPT_ID	DEPT_ID	DEPT_LOC
6200	박철수	200	200	대전
6311	김대정	100	100	서울
7489	민등규	400	400	인천
7522	정종철	400	400	인천
6321	이종길	400	400	인천
6351	진대준	300	300	서울
7910	이영희	300	300	서울
6361	김철수	200	200	대전
7878	백기수	200	200	대전
7854	진영진	400	400	인천
7872	이문정	100	100	서울
7920	김마리아	300	300	서울
7933	김철수	200	200	대전

13 rows selected.

## 2. 동일 조인

### ★ 검색 조건 추가

- 조인 조건 외에 다른 검색 조건을 추가하여 보다 정교한 결과를 얻을 수 있음

### ★ 검색 조건 추가 시 유의사항

- ① 추가된 검색 조건과 조인 조건이 모두 만족되어야 하므로 논리 연산자 'AND'로 연결함
- ② **검색 조건을 조인 조건보다 먼저 기술**하는 것이 효율적임

## 2. 동일 조인

### ✓ 검색 조건이 추가된 동일 조인의 예

- 이름이 '김철수'인 모든 사원의 사원번호와 이름, 담당업무, 부서번호, 근무처 검색하기

```
SELECT emp_id, emp_name, job, emp.dept_id, dept_loc
FROM emp, dept
WHERE emp_name = '김철수'
AND emp.dept_id = dept.dept_id;
```

EMP_ID	EMP_NAME	JOB	DEPT_ID	DEPT_LOC
6361	김철수	부장	200	대전
7933	김철수	사무직	200	대전

## 2. 동일 조인

### ★ 테이블 별칭(Alias)을 사용한 조인

- 질의문 작성을 간소화하기 위해 조인할 각 테이블에 간단한 별칭을 부여하여 사용할 수 있음

```
SELECT  [테이블별칭1.]열이름1, [테이블별칭2.]열이름2, ...  
FROM    테이블이름1 테이블별칭1, 테이블이름2 테이블별칭2  
WHERE   테이블별칭1.열이름3 = 테이블별칭1.열이름4 ;
```

### ★ 테이블 별칭 사용 시 유의사항

- FROM 절에 테이블 별칭을 지정한 경우, SELECT 절이나 WHERE 절에 테이블 별칭 대신 테이블 이름을 사용하면 오류가 발생함

## 2. 동일 조인

### ✓ 테이블 별칭을 사용한 조인의 예

- 급여가 2500 이상인 모든 사원의 사원번호와 이름, 부서번호, 그리고 급여 검색하기

```
SELECT emp_id, emp_name, e.dept_id, sal
FROM emp e, dept d
WHERE sal >= 5000
AND e.dept_id = d.dept_id;
```

EMP_ID	EMP_NAME	DEPT_ID	SAL
6200	박철수	200	9000
6311	김대정	100	6200
6321	이종길	400	7000
6351	진대준	300	5850
6361	김철수	200	6500
7878	백기수	200	5000

6 rows selected.

## 2. 동일 조인

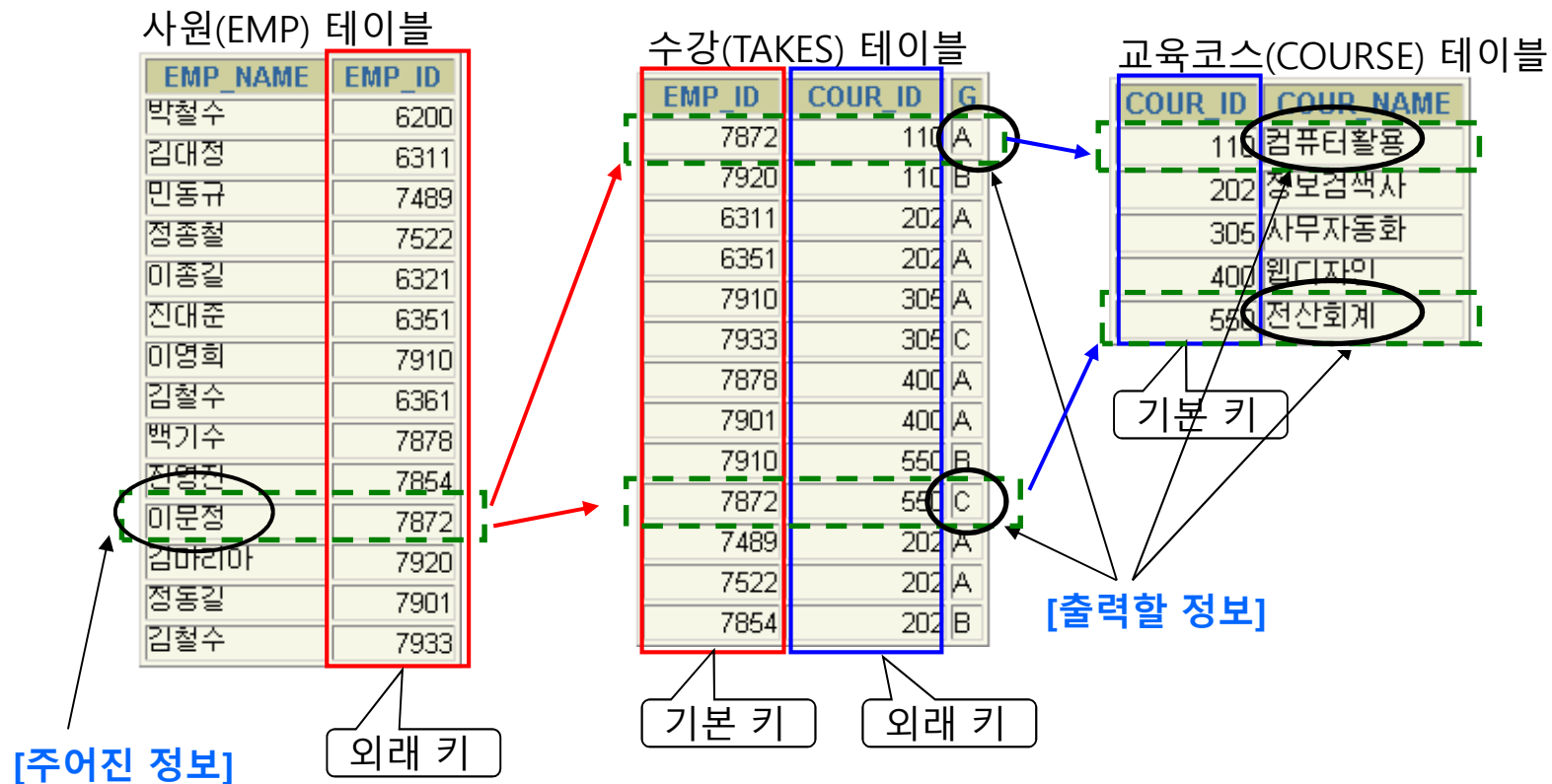
### ★ 두 개 이상의 테이블 조인 시 유의사항

- ① 조인하는 테이블이 5개 이상이면 성능이 급속히 저하되므로 피하는 것이 좋음
- ②  $n$ 개 테이블을 조인할 때, 카티션 프로덕트 연산을 피하기 위해서 반드시  $n-1$ 개 이상의 조인 조건을 기술해야 함

## 2. 동일 조인

### ✓ 3개 테이블 조인 예

- '이문정' 사원이 수강한 모든 교육코스의 이름과 성적 검색하기





## 2. 동일 조인

### ✓ 3개 테이블 조인 예

- '이문정' 사원이 수강한 모든 교육코스의 이름과 성적 검색하기

```
SELECT  cour_name  코스명, grade  성적
FROM    emp e, takes t, course c
WHERE   emp_name = '이문정'
AND     e.emp_id = t.emp_id
AND     t.cour_id = c.cour_id;
```

COUR_NAME	GRA
컴퓨터활용	A
전산회계	C

### 3. 비동일 조인

#### ★ 비동일 조인(Non-equi Join)이란?

- 조인할 양쪽 테이블에 있는 어떤 열도 직접 관련되어 있지 않지만, **의미상 간접적으로 관련된 열을 이용해서 테이블을 조인**하는 것

```
SELECT  [테이블이름1.]열이름1, [테이블이름2.]열이름2, ...  
FROM    테이블이름1, 테이블이름2  
WHERE   (비동일 조인 조건);
```

#### ★ 비동일 조인의 특징

- 양쪽 테이블에 직접 관련된 열은 없지만, 의미상 간접적으로 관련되는 열이 있어야 함

### 3. 비동일 조인

#### ✓ 비동일 조인의 예

- 각 사원의 이름과 급여, 그리고 급여가 해당하는 등급 출력하기

[emp 테이블]

EMP_ID	EMP_NAME	SAL
6200	박철수	9000
6311	김대정	6200
7489	민동규	3700
7522	정종철	4520
6321	이종길	7000
6351	진대준	5850
7910	이영희	3200
6361	김철수	6500
7878	백기수	5000
7854	진영진	3500
7872	이문정	2500
7920	김마리아	2650
7901	정동길	4500
7933	김철수	2500

[sal\_class 테이블]

CLASS	LBOUND	HBOUND
5	2000	3000
4	3001	4000
3	4001	5000
2	5001	7000
1	7001	99999

emp 테이블의 sal은 sal\_class  
테이블의 lbound 및 hbound와  
관련성이 있음

[제목] 간접적으로 관련된 두개 테이블

### 3. 비동일 조인

#### ✓ 비동일 조인의 예

- 각 사원의 이름과 급여, 그리고 급여가 해당하는 등급 출력하기

```
SELECT emp_name, sal, class
FROM emp, sal_class
WHERE sal BETWEEN lbound AND hbound;
```

EMP_NAME	SAL	CLASS
이문정	2500	5
김철수	2500	5
김마리아	2650	5
이영희	3200	4
진영진	3500	4
민동규	3700	4
정동길	4500	3
정종철	4520	3
백기수	5000	3
진대준	5850	2
김대정	6200	2
김철수	6500	2
이종길	7000	2
박철수	9000	1

14 rows selected.

## 4. 외부 조인

### ★ 외부 조인(Outer Join)이란?

- 조인할 한쪽 테이블에 조인 조건을 만족하는 행이 없어도, 그 테이블에 '널(Null)' 행을 추가해서 결과 테이블에 포함시키는 연산
- 정보가 부족한 테이블에 널(Null) 행을 추가시키기 위해 외부 조인 연산자 '+'를 사용함

```
SELECT 열이름1, 열이름2, ...  
FROM   테이블이름1, 테이블이름2  
WHERE  테이블이름1.열이름3(+) = 테이블이름2.열이름4;
```

```
SELECT 열이름1, 열이름2, ...  
FROM   테이블이름1, 테이블이름2  
WHERE  테이블이름1.열이름3 = 테이블이름2.열이름4(+);
```

### ★ 외부 조인 시 유의사항

- 반드시 조인할 한쪽 테이블에만 외부 조인 연산자를 지정해야 함

## 4. 외부 조인

### ✓ 외부 조인이 필요한 테이블 사례

EMP 테이블

EMP_NAME	DEPT_ID
박철수	200
김대정	100
민동규	400
정종철	400
이종길	400
진대준	300
이영희	300
김철수	200
백기수	200
진영진	400
이문정	100
김마리아	300
정동길	
김철수	200

DEPT 테이블

DEPT_ID	DEPT_NAME
100	관리부
200	연구부
300	총무부
400	영업부
500	개발부

500번 개발부는 신설된  
부서로, 현재 사원이 없음

부서번호가 Null인 행이 없으므로  
EMP 테이블을 기준으로 생각하면  
정보가 부족함

신입 사원이라 아직  
부서가 정해지지 않았음

500번 부서(개발부)에서 근무하는  
사원이 없으므로, DEPT 테이블을  
기준으로 생각하면 정보가 부족함

- 대응하는 정보가 부족한 경우, 원하는 결과에 따라 외부 조인 필요함

## 4. 외부 조인

### ✓ 외부 조인 예(1)

- 모든 사원의 이름과 부서번호, 부서이름을 부서번호의 오름차순으로 출력하되, 아직 근무자가 없는 **신설 부서의 부서번호와 부서이름도 출력하기**

```
SELECT emp_name, d.dept_id, dept_name
FROM emp e, dept d
WHERE e.dept_id(+) = d.dept_id
ORDER BY e.dept_id;
```

EMP_NAME	DEPT_ID	DEPT_NAME
김대정	100	관리부
이문정	100	관리부
김철수	200	연구부
김철수	200	연구부
백기수	200	연구부
박철수	200	연구부
진대준	300	총무부
이영희	300	총무부
김마리아	300	총무부
진영진	400	영업부
정종철	400	영업부
민동규	400	영업부
이종길	400	영업부
	500	개발부

14 rows selected.

## 4. 외부 조인

### ✓ 외부 조인 예(2)

- 모든 사원의 이름과 부서번호, 부서이름을 부서번호의 오름차순으로 출력하되, **부서가 정해지지 않은 신입 사원의 이름도 출력하기**

```
SELECT emp_name, d.dept_id, dept_name
FROM emp e, dept d
WHERE e.dept_id = d.dept_id(+)
ORDER BY e.dept_id;
```

EMP_NAME	DEPT_ID	DEPT_NAME
이문정	100	관리부
김대정	100	관리부
김철수	200	연구부
백기수	200	연구부
김철수	200	연구부
박철수	200	연구부
김마리아	300	총무부
진대준	300	총무부
이영희	300	총무부
진영진	400	영업부
이종길	400	영업부
민동규	400	영업부
정종철	400	영업부
정동길		

14 rows selected.



## 5. 자체 조인

### ★ 자체 조인(Self Join)이란?

- 물리적으로 하나인 테이블에서 논리적으로 다른 테이블로 간주하고 조인하는 연산

```
SELECT 열이름1, 열이름2, ...  
FROM   테이블이름1 테이블별칭1, 테이블이름1 테이블별칭2  
WHERE  (테이블별칭을 사용한 자체 조인 조건);
```

### ★ 자체 조인 시 유의사항

- 조인할 동일한 테이블에 서로 다른 별칭을 부여해야 함

### ★ 자체 조인의 필요성

- 원하는 정보가 한 개 테이블 내에 존재하지만, 직접 검색할 수는 없는 경우

## 5. 자체 조인

### ✓ 자체 조인이 필요한 테이블 사례

- 사원의 관리자 이름을 검색할 때

EMP 테이블(사원 : WORKER)

EMP_ID	EMP_NAME	MGR_ID
6200	박철수	
6311	김대정	6200
7489	민동규	6321
7522	정종철	6321
6321	이종길	6200
6351	진대준	6200
7910	이영희	6351
6361	김철수	6200
7878	백기수	6361
7854	진영진	6321
7872	이문정	6311
7920	김마리아	6351
7901	정동길	
7933	김철수	6361

EMP 테이블(관리자 : MANAGER)

EMP_ID	EMP_NAME	JOB
6200	박철수	대표이사
6311	김대정	부장
6321	이종길	부장
6351	진대준	부장
6361	김철수	부장

기본 키

참조 키

[제목] 대응하는 정보가 동일한 테이블에 있는 경우

## 5. 자체 조인

### ✓ 자체 조인의 예

- 각 사원의 이름과 직속 관리자의 이름 출력하기

```
SELECT worker.emp_name 사원, manager.emp_name 관리자
FROM emp worker, emp manager
WHERE worker.mgr_id = manager.emp_id;
```

사원	관리자
김철수	박철수
진대준	박철수
이종길	박철수
김대정	박철수
이문정	김대정
진영진	이종길
정종철	이종길
민동규	이종길
김마리아	진대준
이영희	진대준
김철수	김철수
백기수	김철수

12 rows selected.



**Q & A**