

# 데이터베이스관리

12주차

담당교수: 김희숙  
(jasmin11@hanmail.net)

# [Quiz] SELECT (노트 작성)

(노트에 풀이하면서 공부하세요)

담당교수: 김희숙  
(jasmin11@hanmail.net)

## [실습] 합집합



[실습]

-- department(deptno, deptname, floor)

-- employee(empno, empname, title, manager, salary, dno)

-- SQL문법을 작성하시오

-- 16) 합집합

deptno	deptname	floor
1	영업	8
2	기획	10
3	개발	9
4	총무	7

empno	empname	title	manager	salary	dno
1003	조민희	과장	4377	3000000	2
1365	김상원	사원	3426	1500000	1
2106	김창섭	대리	1003	2500000	2
3011	이수민	부장	4377	4000000	3
3426	박영권	과장	4377	3000000	1
3427	최종철	사원	3011	1500000	3
4377	이성래	이사	NULL	5000000	2

-- 16-1) 김창섭이 속한 부서번호와 부서의 모든 부서번호를 합집합으로 검색하라 (UNION)

-- 16-2) 김창섭이 속한 부서번호와 부서의 모든 부서번호를 합집합으로 검색하라 (UNION ALL)

## [실습] 조인



[실습]

```
-- department(deptno, deptname, floor)
-- employee(empno, empname, title, manager, salary, dno)
```

-- **노트에 SQL문법을 작성하시오**

-- 17) 조인

-- 19) 조인 + 정렬

deptno	deptname	floor
1	영업	8
2	기획	10
3	개발	9
4	총무	7

empno	empname	title	manager	salary	dno
1003	조민희	과장	4377	3000000	2
1365	김상원	사원	3426	1500000	1
2106	김창섭	대리	1003	2500000	2
3011	이수민	부장	4377	4000000	3
3426	박영권	과장	4377	3000000	1
3427	최종철	사원	3011	1500000	3
4377	이성래	이사	NULL	5000000	2

-- 17) 사원의 이름과 이 사원이 속한 부서이름을 검색하라 (조인)

-- 19) 사원에 대해서 부서이름, 사원이름, 직급, 급여를 검색하라.

-- 부서이름에 대해서 오름차순, 부서이름이 같을 경우에는 salary에 대해서 내림차순으로 정렬하라

## [실습] 자체 릴레이션(emp)

[실습]

-- emp(empno, empname, manager, dno)

[예제] 사원명과 직속상사이름을 검색하라

emp 테이블					--실행결과	
	empno	empname	manager	dno		
1	1003	조민희	3011	1		사원명 직속상사명
2	2106	김창섭	3426	2	1	조민희 이수민
3	3011	이수민	NULL	1	2	김창섭 박영권
4	3426	박영권	3011	3	3	박영권 이수민
5	3427	최종철	2106	3	4	최종철 김창섭

--풀이과정

emp E					emp M				
	empno	empname	manager	dno		empno	empname	manager	dno
1	1003	조민희	3011	1	1	1003	조민희	3011	1
2	2106	김창섭	3426	2	2	2106	김창섭	3426	2
3	3011	이수민	NULL	1	3	3011	이수민	NULL	1
4	3426	박영권	3011	3	4	3426	박영권	3011	3
5	3427	최종철	2106	3	5	3427	최종철	2106	3

--SQL 답안을 작성하시오



-- **노트**에 SQL문법을 작성하시오

-- **사원명과 직속상사명을 검색하라**

-- [실습] 사원명과 직속상사명을 검색하라

# 관계대수-02

실렉션, 프로젝션, 조인

담당교수: 김희숙  
(jasmin11@hanmail.net)

## [예제] [Quiz]

노트  
풀이

[Quiz 01] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

R1	A	B
	1	4
	2	5

S1	A	C	D
	1	3	4
	2	2	6
	3	1	9

다음 결과는?  
1.  $R1 \times S1$   
2.  $R1 \bowtie S1$   
3.  $R1 \bowtie_N S1$   
4.  $R1 \bowtie_{A \geq C} S1$

-- **노트**에 각 실행결과를 작성하시오

- 1. 카티션 프로덕트
- 2. 동등조인
- 3. 자연조인
- 4. 세타조인

## [예제] [Quiz]

[Quiz 01] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

R1	A	B
	1	4
	2	5

S1	A	C	D
	1	3	4
	2	2	6
	3	1	9

다음 결과는?

1. R1 X S1

2. R1 ⋈ S1

3. R1 ⋈<sub>N</sub> S1

4. R1 ⋈<sub>A >= C</sub> S1

-- 1. 카티션 프로덕트

select \*

from R1, S1;

-- 2. 동등조인(equi join)

select \*

from R1, S1

where R1.A = S1.A;



## [예제] [Quiz]

[Quiz 01] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

R1	<table><tr><th>A</th><th>B</th></tr><tr><td>1</td><td>4</td></tr><tr><td>2</td><td>5</td></tr></table>	A	B	1	4	2	5	S1	<table><tr><th>A</th><th>C</th><th>D</th></tr><tr><td>1</td><td>3</td><td>4</td></tr><tr><td>2</td><td>2</td><td>6</td></tr><tr><td>3</td><td>1</td><td>9</td></tr></table>	A	C	D	1	3	4	2	2	6	3	1	9	다음 결과는?
A	B																					
1	4																					
2	5																					
A	C	D																				
1	3	4																				
2	2	6																				
3	1	9																				
1. R1 X S1																						
2. R1 ⋈ S1																						
3. R1 ⋈ <sub>N</sub> S1																						
4. R1 ⋈ <sub>A &gt;= C</sub> S1																						

-- 2. 동등조인(equi join)

select \*

from R1, S1

where R1.A = S1.A;

-- 3. 자연조인(natural join)

select R1.A, B, C, D

from R1, S1

where R1.A = S1.A;

## [예제] [Quiz]

[Quiz 01] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

R1	A	B
	1	4
	2	5

S1	A	C	D
	1	3	4
	2	2	6
	3	1	9

다음 결과는?

1. R1 X S1

2. R1 ⋈ S1

3. R1 ⋈<sub>N</sub> S1

4. R1 ⋈<sub>A >= C</sub> S1

-- 관계 연산자

>, >=, =, !=, <=, <

-- 4. 세타조인(theta join)

select \*

from R1, S1

where R1.A >= C;

# [복습] 조인(join)

(내부조인, 외부조인)

담당교수: 김희숙  
(jasmin11@hanmail.net)

## [예제] [Quiz]



[복습] (조인) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

[내부조인] (테이블 2개) freshman, dmember

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

-- **노트**에 각 실행결과를 작성하시오

- 1. 카티션 프로젝트
- 2. 동등조인
- 3. 자연조인
  
- 4. 왼쪽외부조인
- 5. 오른쪽외부조인
- 6. 완전외부조인



[Quiz 02] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

관계대수 / SQL

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

1. 카티션 프로덕트(product)

2. 동등조인(equi join)

3. 자연조인(natural join)

-- **노트**에 각 실행결과를 작성하시오

-- 1. 카티션 프로덕트

-- 2. 동등조인

-- 3. 자연조인

1-1) 관계대수

1-2) SQL

1-3) 실행결과

[Quiz 02] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

관계대수 / SQL

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

- 1-1) 관계대수
- 1-2) SQL
- 1-3) 실행결과

-- **노트**에 각 실행결과를 작성하시오

- 4. 왼쪽외부조인
- 5. 오른쪽외부조인
- 6. 완전외부조인

- 4. 왼쪽외부조인(left outer join)
- 5. 오른쪽외부조인(right outer join)
- 6. 완전외부조인(full outer join)

## [예제] [Quiz]

[Quiz 02] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

관계대수 / SQL

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

- 1-1) 관계대수
- 1-2) SQL
- 1-3) 실행결과

-- 2. 동등조인(equi join)

```
select *  
from freshman, dmember  
where freshman.name=dmember.name;
```

-- 2. 동등조인(equi join)

```
select *  
from freshman INNER JOIN dmember  
on freshman.name=dmember.name;
```

-- 1. 카티션 프로덕트

```
select *  
from freshman, dmember;
```

-- 3. 자연조인(natural join)

```
select freshman.name, address, dept_name  
from freshman, dmember  
where freshman.name=dmember.name;
```

-- 3. 자연조인(natural join)

```
select freshman.name, address, dept_name  
from freshman INNER JOIN dmember  
on freshman.name=dmember.name;
```

## [예제] [Quiz]

[Quiz 02] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

관계대수 / SQL

freshman

name	address
김광식	서울
김현정	대전
조영수	대전

dmember

name	dept_name
김광식	컴퓨터공학과
김현정	산업공학과
이진영	전자공학과

1-1) 관계대수

1-2) SQL

1-3) 실행결과

-- 4. 왼쪽 외부조인(left outer join)

```
select freshman.name, address, dept_name
from  freshman LEFT OUTER JOIN dmember
on    freshman.name=dmember.name;
```

-- 5. 오른쪽 외부조인(right outer join)

```
select freshman.name, address, dept_name
from  freshman RIGHT OUTER JOIN dmember
on    freshman.name=dmember.name;
```

-- 6. 완전 외부조인(full outer join)



# [Quiz] 조인(join)

(내부조인, 외부조인)

담당교수: 김희숙  
(jasmin11@hanmail.net)

## [예제] [Quiz]



[Quiz 1-1] (조인) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

omember

id	groupid
100	A
101	B
102	A
103	F

group\_ex

groupid	position
A	서울
B	대구
C	광주
D	부산
E	대전

-- **노트**에 각 실행결과를 작성하시오

- 1. 카티션 프로덕트
- 2. 동등조인
- 3. 자연조인
- 4. 왼쪽외부조인
- 5. 오른쪽외부조인
- 6. 완전외부조인

## [예제] [Quiz]

[Quiz 1-2] (조인) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라



emp	ename	pno
	홍길동	101
	임꺽정	102
	박찬호	101
	박찬호	103
	신동엽	NULL

project	pno	pname
	101	작전중
	102	특공대
	103	유레카
	104	다모여

-- **노트**에 각 실행결과를 작성하시오

- 1. 카티션 프로젝트
- 2. 동등조인
- 3. 자연조인
- 4. 왼쪽외부조인
- 5. 오른쪽외부조인
- 6. 완전외부조인

## [예제] [Quiz]



[Quiz 1-3] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

R	A	B	C
	a1	b1	c1
	a2	b1	c1
	a3	b1	c2
	a4	b2	c3

S	B	C	D
	b1	c1	d1
	b1	c1	d2
	b2	c2	d3

-- **노트**에 각 관계대수식과 실행결과를 작성하시오

-- 1) 관계대수식으로 작성하시오

```
SELECT R.B, R.C FROM R
UNION
SELECT S.B, S.C FROM S;
```

-- 2) 관계대수식으로 작성하시오

```
SELECT R.B, R.C FROM R
INTERSECT
SELECT S.B, S.C FROM S;
```

-- 3) 관계대수식으로 작성하시오

```
SELECT R.B, R.C FROM R
MINUS
SELECT S.B, S.C FROM S;
```

-- 4) 관계대수식으로 작성하시오

```
SELECT *
FROM R, S;
```

# 관계대수-02

(관계대수)

담당교수: 김희숙  
(jasmin11@hanmail.net)

## [실습] 관계대수

### <성적>

이름	국어	영어	수학	과학
고길동	70	90	78	88
이순신	90	80	88	98
둘리	80	68	98	78
아무개	79	79	70	67
임꺽정	67	98	80	84

```
CREATE TABLE 성적 (
  이름 varchar(20) ,
  국어 int ,
  영어 int ,
  수학 int ,
  과학 int ,
  PRIMARY KEY(이름)
);
```

```
insert into 성적 values('고길동',70,90,78,88);
insert into 성적 values('이순신',90,80,88,98);
insert into 성적 values('둘리',80,68,98,78);
insert into 성적 values('아무개',79,79,70,57);
insert into 성적 values('임꺽정',67,98,80,84);
```

## [실습] 관계대수(quiz\_sanup\_algebra.sql)

### <학적부>

이름	주소	나이
고길동	모진동	25
이순신	화양동	20
둘리	자양동	24
아무개	화양동	19
임꺽정	구의동	18

```
CREATE TABLE 학적부 (
  이름 varchar(20) ,
  주소 varchar(50) ,
  나이 int ,
  PRIMARY KEY(이름) ,
  FOREIGN KEY(이름) REFERENCES 성적(이름)
);
```

```
insert into 학적부 values('고길동','모진동',25);
insert into 학적부 values('이순신','화양동',20);
insert into 학적부 values('둘리','자양동',24);
insert into 학적부 values('아무개','화양동',19);
insert into 학적부 values('임꺽정','구의동',18);
```

## [예제] 관계대수

### <성적>

이름	국어	영어	수학	과학
고길동	70	90	78	88
이순신	90	80	88	98
둘리	80	68	98	78
아무개	79	79	70	57
임꺽정	67	98	80	84

### <학적부>

이름	주소	나이
고길동	모진동	25
이순신	화양동	20
둘리	자양동	24
아무개	화양동	19
임꺽정	구의동	18

**예제 1**  $\sigma_{\text{국어} \geq 80}(\text{성적})$  : <성적> 릴레이션에서 국어 점수가 80점 이상인 튜플들을 추출한다.

이름	국어	영어	수학	과학
이순신	90	80	88	98
둘리	80	68	98	78

```
SELECT *
FROM 성적
WHERE 국어 >= 80;
```

**예제 2**  $\sigma_{\text{국어} \geq 80 \vee \text{영어} \geq 80}(\text{성적})$  : <성적> 릴레이션에서 국어 점수 또는 영어 점수가 80점 이상인 튜플들을 추출한다.

이름	국어	영어	수학	과학
고길동	70	90	78	88
이순신	90	80	88	98
둘리	80	68	98	78
임꺽정	67	98	80	84

```
SELECT *
FROM 성적
WHERE 국어 >= 80 OR 영어 >= 90;
```

## [예제] 관계대수

<성적>

이름	국어	영어	수학	과학
고길동	70	90	78	88
이순신	90	80	88	98
둘리	80	68	98	78
아무개	79	79	70	57
임꺽정	67	98	80	84

<학적부>

이름	주소	나이
고길동	모진동	25
이순신	화양동	20
둘리	자양동	24
아무개	화양동	19
임꺽정	구의동	18

**예제**  $\pi_{\text{이름, 국어}}(\text{성적})$  : <성적> 릴레이션에서 이름과 국어 속성을 추출한다,

이름	국어
고길동	70
이순신	90
둘리	80
아무개	79
임꺽정	67

```
SELECT 이름, 국어
FROM 성적;
```



## [예제] 관계대수

**예제** 성적 $\bowtie$ 이름=이름학적부 : <성적> 릴레이션과 <학적부> 릴레이션으로부터 이름 속성이 같은 튜플들을 결합하여 새로운 릴레이션을 생성한다.

<성적>

이름	국어	영어	수학	과학
고길동	70	90	78	88
이순신	90	80	88	98
둘리	80	68	98	78
아무개	79	79	70	57
임꺽정	67	98	80	84

<학적부>

이름	주소	나이
고길동	모진동	25
이순신	화양동	20
둘리	자양동	24
아무개	화양동	19
임꺽정	구의동	18



성적,이름	국어	영어	수학	과학	학적부,이름	주소	나이
고길동	70	90	78	88	고길동	모진동	25
이순신	90	80	88	98	이순신	화양동	20
둘리	80	68	98	78	둘리	자양동	24
아무개	79	79	70	57	아무개	화양동	19
임꺽정	67	98	80	84	임꺽정	구의동	18

## [실습] 관계대수(Oracle)

<사원>

사번	이름
1	고길동
2	이순신

<직원>

사번	이름
2	이순신
3	둘리

**예제1**  $\pi_{\text{이름}}(\text{사원}) \cup \pi_{\text{이름}}(\text{직원}) :$

이름
고길동
이순신
둘리

**예제2**  $\pi_{\text{이름}}(\text{사원}) \cap \pi_{\text{이름}}(\text{직원}) :$

이름
이순신

**예제3**  $\pi_{\text{이름}}(\text{사원}) - \pi_{\text{이름}}(\text{직원}) :$

이름
고길동

```
select * from 사원
UNION
select * from 직원;
```

```
select * from 사원
INTERSECT
select * from 직원;
```

```
select * from 사원
MINUS
select * from 직원;
```

## [실습]

```
CREATE TABLE 사원 (
  사번 int ,
  이름 varchar(20) ,
  PRIMARY KEY(사번)
);
```

```
CREATE TABLE 직원 (
  사번 int ,
  이름 varchar(20) ,
  PRIMARY KEY(사번)
);
```

```
insert into 사원 values(1, '고길동');
insert into 사원 values(2, '이순신');
```

```
insert into 직원 values(2, '이순신');
insert into 직원 values(3, '둘리');
```

## [예제] 관계대수: [문제 1]

<학생> 릴레이션

학번	이름	학년	학과
1001	나기태	4	컴퓨터
1002	송찬영	3	전기
1003	정연	1	컴퓨터
1004	박병화	4	건축
1005	김미나	1	컴퓨터

<학생> 릴레이션을 대상으로 <예1>과 같은 릴레이션을 생성하는 (① ) 연산의 표기형식은 (② )와 같이 나타낼 수 있다.

<예1>

학번	이름	학년	학과
1001	나기태	4	컴퓨터
1004	박병화	4	건축

답항보기

1	JOIN	2	UNION	3	PROJECT	4	$\pi$ 학년, 학과(학생)	5	DIFFERENCE
6	CARTESIAN	7	$\sigma$ 학년=4(학생)	8	SELECT	9	제약(Constraint)	10	DIVISION

## [예제] 관계대수: [문제 1]

순수 관계 연산들 중 (③ )연산은 릴레이션의 애트리뷰트 들을 대상으로 하는 연산으로, 연산에 명시된 애트리뷰트 값들만 선택한다. (③ )연산의 결과로 만들어진 릴레이션은 주어진 릴레이션의 몇몇 열들로 구성된 것으로 릴레이션의 수직적 부분 집합과 같으므로 수직연산이라고도 한다. (③ )연산의 결과 릴레이션은 중복된 튜플을 배제하고 생성된다. 예를들어, <학생>릴레이션을 대상으로 <예2>와 같은 결과 릴레이션을 생성하는 (③ )연산의 표기 형식은 (④ )(으)로 나타낼 수 있다.

<예2>

학번	학과
4	컴퓨터
3	전기
1	컴퓨터
4	건축

관계대수 연산을 이용하여 <예3>과 같이 <학생>릴레이션에 대해 학과가 “컴퓨터”이고, 학년이 1인 학생의 학번과 이름으로 구성된 릴레이션 표기형식은  $\pi_{\text{학번}, \text{이름}}(\sigma_{\text{학과}=\text{“컴퓨터”} \wedge \text{학년}=1}(\text{학생}))$ 으로 나타낼 수 있다.

<예3>

학번	이름
1003	정연
1005	김미나

답항보기

1	JOIN	2	UNION	3	PROJECT	4	$\pi_{\text{학번}, \text{학과}}(\text{학생})$	5	DIFFERENCE
6	CARTESIAN	7	$\sigma_{\text{학번}=4}(\text{학생})$	8	SELECT	9	제약(Constraint)	10	DIVISION

## [예제] 관계대수: [문제 1]

또한, 순수 관계 연산에는 공통 속성을 중심으로 2개의 릴레이션을 하나로 합쳐서 새로운 릴레이션을 만드는 (⑤ )연산 등이 있고, (⑤ )연산의 연산자 기호는 " $\bowtie$ "를 사용한다.

### 답항보기

1	JOIN	2	UNION	3	PROJECT	4	$\pi$ 학년,학과(학생)	5	DIFFERENCE
6	CARTESIAN	7	$\sigma$ 학년=4(학생)	8	SELECT	9	제약(Constraint)	10	DIVISION



## [실습] 관계대수

```
CREATE TABLE department (
  deptno int ,
  deptname varchar(20) ,
  PRIMARY KEY(deptno)
);

CREATE TABLE employee (
  empno char(4) ,
  empname varchar(20) ,
  dno int
  PRIMARY KEY(empno) ,
  FOREIGN KEY(dno) REFERENCES department(deptno)
);
```

EMPLOYEE

EMPNO	EMPNAME	DNO
1234	김원호	2
1235	박영주	1
1236	이수미	3
1237	조민수	2
1238	최종인	3

## [실습] 관계대수(quiz\_sanup\_algebra.sql)

```
insert into department values(1,'영업');
insert into department values(2,'기획');
insert into department values(3,'개발');
insert into department values(4,'총무');

insert into employee values('1234','김원호',2);
insert into employee values('1235','박영주',1);
insert into employee values('1236','이수미',3);
insert into employee values('1237','조민수',2);
insert into employee values('1238','최종인',3);
```

DEPARTMENT

DEPTNO	DEPTNAME
1	영업
2	기획
3	개발
4	총무

'EMPLOYEE(④ ) DNO=DEPTNO DEPARTMENT' 로 동등 조인한 결과는 다음과 같다.

EMPNO	EMPNAME	DNO	DEPTNO	DEPTNAME
1234	김원호	2	2	영업
1235	박영주	1	1	기획
1236	이수미	3	3	개발
1237	조민수	2	2	총무
1238	최종인	3	3	개발

# SQL: 조인(join)

(내부조인, 외부조인)

담당교수: 김희숙  
(jasmin11@hanmail.net)

# 조인(join)



## □조인

- ✓ 2개 이상의 테이블로 부터 원하는 결과를 나타내는 것
- ✓ 2개 이상의 테이블로부터 필요한 열을 선택하여 하나의 테이블로 논리적인 결합하는 것

개인신상

이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소
박찬호	730201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2
선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2
임꺽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2
차범근	600809-1987766	1960-08-09	친구	(019) 234-2342	(555-555)	서울시 종로구 관철동 234-2
홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24

자동차

이름	차종	차명	년식	구입일
임꺽정	트럭	포토	95	97-5
차범근	승용차	에쿠스	01	01-8
홍길동	승용차	소나타	98	98-8



이름	생년월일	차종	차명
임꺽정	1970-01-01	트럭	포토
차범근	1960-08-09	승용차	에쿠스
홍길동	1965-12-14	승용차	소나타



## [실습] 조인(vehicle\_join-mysql)

### (테이블 1개) 고객관리

#### ❖ 테이블 1개인 경우

	이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소	차종	차명	년식	구입일
▶	박찬호	720201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2	NULL	NULL	NULL	NULL
	선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2	NULL	NULL	NULL	NULL
	임궽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2	트럭	포토	1995	1997-05
	차범근	600809-1987766	1960-08-09	친구	(017) 234-2342	(555-555)	서울시 종로구 관철동 234-2	승용차	메쿠스	2001	2001-08
	홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24	승용차	소나타	1998	1998-08

### (테이블 2개) 개인신상, 자동차

#### ❖ 테이블 2개인 경우

	이름	주민번호	생년월일	관계	핸드폰번호	우편번호	주소
▶	박찬호	720201-1034343	1973-02-01	후배	(017) 234-2342	(333-333)	서울시 강남구 역삼동 234-2
	선동열	680709-1078656	1968-07-09	동생	(018) 234-2342	(444-444)	서울시 서초구 서초동 234-2
	임궽정	700101-1027362	1970-01-01	고객	(016)2342-2342	(222-222)	서울시 영등포구 여의도동 234-2
	차범근	600809-1987766	1960-08-09	친구	(017) 234-2342	(555-555)	서울시 종로구 관철동 234-2
	홍길동	651214-1078767	1965-12-14	친구	(011) 234-2342	(111-111)	서울시 구로구 구로본동 234-24

	이름	생년월일	차종	차명
▶	임궽정	1970-01-01	트럭	포토
	차범근	1960-08-09	승용차	메쿠스
	홍길동	1965-12-14	승용차	소나타

# [요약] 조인

- ❖ 내부조인(theta join, equi join, natural join)
- ❖ 외부조인(left outer join, right outer join, full outer join)

## 1. 내부조인:

-- 방법1)

SELECT ..

FROM ..

WHERE ..

AND ...

AND ...

AND ...

## 1. 내부조인:

-- 방법2)

SELECT ..

FROM .. INNER JOIN ...

ON ..

WHERE ...

AND ...

AND ...

## [실습] (실습 01) 조인

고객

고객아이디	고객이름	나이	등급
apple	김현준	20	gold
banana	정소화	25	vip
carrot	원유선	28	gold
orange	정지영	22	silver

주문

주문번호	주문고객	주문제품	수량
1001	apple	진짜우동	10
1002	carrot	맛있는파이	5
1003	banana	그대로만두	11

### [실습]

- 고객(고객아이디, 고객이름, 나이, 등급)
- 주문(주문번호, 주문고객, 주문제품, 수량)

- (자연조인)
- 1-1) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라

- (왼쪽 외부조인)
- 1-2) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라
- (단, 주문하지 않은 고객도 포함)

## [실습] (실습 1-01) 내부조인

고객 릴레이션

고객아이디	고객이름	나이
apple	김현준	20
banana	정소화	25
carrot	원유선	28
orange	정지영	22

주문 릴레이션

주문번호	주문 고객	주문제품
1001	apple	진짜우동
1002	carrot	맛있는파이
1003	banana	그대로만두

↓ 자연 조인 연산

고객 ⋈<sub>N</sub> 주문

고객아이디	고객이름	나이	주문번호	주문제품
apple	김현준	20	1001	진짜우동
banana	정소화	25	1003	그대로만두
carrot	원유선	28	1002	맛있는파이

-- (자연조인)  
 -- 1-1) 고객아이디, 고객이름, 나이,  
 -- 주문번호, 주문제품을 검색하라

-- 1-1) 고객아이디, 고객이름, 나이,  
 -- 주문번호, 주문제품을 검색하라  
 SELECT 고객.고객아이디, 고객이름, 나이, 주문번호, 주문제품  
 FROM 고객, 주문  
 WHERE 고객.고객아이디 = 주문.주문고객;

## [실습] (실습 1-02) 외부조인

고객

고객아이디	고객이름	나이
apple	김현준	20
banana	정소화	25
carrot	원유선	28
orange	정지영	22

주문

주문번호	주문고객	주문제품
1001	apple	진짜우동
1002	carrot	맛있는파이
1003	banana	그대로만두

원쪽 외부 조인 연산

고객 ⋈ 주문

고객아이디	고객이름	나이	주문번호	주문제품
apple	김현준	20	1001	진짜우동
banana	정소화	25	1003	그대로만두
carrot	원유선	28	1002	맛있는파이
orange	정지영	22	NULL	NULL

-- (왼쪽 외부조인)  
 -- 1-2) 고객아이디, 고객이름, 나이,  
 -- 주문번호, 주문제품을 검색하라  
 -- (단, 주문하지 않은 고객도 포함)

-- (왼쪽 외부조인)  
 -- 1-2) 고객아이디, 고객이름, 나이,  
 -- 주문번호, 주문제품을 검색하라  
 -- (단, 주문하지 않은 고객도 포함)  
 SELECT 고객.고객아이디, 고객이름, 나이, 주문번호, 주문제품  
 FROM 고객 LEFT OUTER JOIN 주문  
 ON 고객.고객아이디 = 주문.주문고객;

## [실습] (실습 01) 조인

고객

고객아이디	고객이름	나이	등급
apple	김현준	20	gold
banana	정소화	25	vip
carrot	원유선	28	gold
orange	정지영	22	silver

주문

주문번호	주문고객	주문제품	수량
1001	apple	진짜우동	10
1002	carrot	맛있는파이	5
1003	banana	그대로만두	11

### [실습]

- 고객(고객아이디, 고객이름, 나이, 등급)
- 주문(주문번호, 주문고객, 주문제품, 수량)

- (자연조인)
- 1-1) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라

- (왼쪽 외부조인)
- 1-2) 고객아이디, 고객이름, 나이,
- 주문번호, 주문제품을 검색하라
- (단, 주문하지 않은 고객도 포함)



## [실습] (실습 2) 내부조인

### [실습]

```
-- department(deptno, deptname, floor)
-- employee(empno, empname, title, manager, salary, dno)
```

demartment

deptno	deptname	floor
1	영업	8
2	기획	10
3	개발	9
4	총무	7

employeee

empno	empname	title	manager	salary	dno
1003	조민희	과장	4377	3000000	2
1365	김상원	사원	3426	1500000	1
2106	김창섭	대리	1003	2500000	2
3011	이수민	부장	4377	4000000	3
3426	바경ㅇ권	과장	4377	3000000	1
3427	최종철	사원	3011	1500000	3
4377	이성래	이사	NULL	5000000	2

- 16-1) 김창섭이 속한 부서번호와 부서의 모든 부서번호를 합집합으로 검색하라 (UNION)
- 16-2) 김창섭이 속한 부서번호와 부서의 모든 부서번호를 합집합으로 검색하라 (UNION ALL)
- 17) 사원의 이름과 이 사원이 속한 부서이름을 검색하라 (조인)
- 19) 사원에 대해서 부서이름, 사원이름, 직급, 급여를 검색하라.
- 부서이름에 대해서 오름차순, 부서이름이 같을 경우에는 salary에 대해서 내림차순으로 정렬하라

# [실습] 조인(테이블 3개)

[내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

- 3-1) 수강한 학생의 학생명, 과목명, 성적을 검색하라(내부조인)
- 3-2) 이홍근의 학생명, 과목명, 성적을 검색하라
- 3-3) 전체 학생의 학생명, 과목명, 성적을 검색하라(외부조인)



## [실습] 조인(테이블 3개)

### [내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

-- 3-1) 수강한 학생의 학생명, 과목명, 성적을 검색하라(내부조인)

```
SELECT 학생명, 과목명, 성적
FROM 학생, 수강, 과목
WHERE 학생.학번 = 수강.학번
      AND 수강.과목번호 = 과목.과목번호;
```

```
SELECT 학생명, 과목명, 성적
FROM 학생 INNER JOIN 수강
ON 학생.학번 = 수강.학번
   INNER JOIN 과목
ON 수강.과목번호 = 과목.과목번호;
```

## [실습] 조인(테이블 3개)

### [내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

```
-- 3-2) 이홍근의 학생명, 과목명, 성적을 검색하라
SELECT 학생명, 과목명, 성적
FROM 학생, 수강, 과목
WHERE 학생.학번 = 수강.학번
AND 수강.과목번호 = 과목.과목번호
AND 학생명 = '이홍근';
```

```
SELECT 학생명, 과목명, 성적
FROM 학생 INNER JOIN 수강
ON 학생.학번 = 수강.학번
INNER JOIN 과목
ON 수강.과목번호 = 과목.과목번호
WHERE 학생명 = '이홍근';
```

## [실습] 조인(테이블 3개)

### [내부조인] (테이블 3개) 학생, 수강, 과목

(3개 테이블 조인)

학생

학번	학생명
11002	이홍근
24036	김순미
30419	박상웅
30555	홍서범
31890	유동희
31892	이정무
32000	김윤식

수강

학번	과목번호	성적
11002	CS310	98
11002	CS313	88
24036	CS345	90
30419	CS326	78
31892	CS388	86
32000	CS378	94
31890	CS310	92

과목

과목번호	과목명
CS310	데이터베이스
CS313	운영체제
CS345	자료구조
CS326	자바
CS546	DB프로그래밍
CS378	멀티미디어
CS388	웹프로그래밍

-- 3-3) 전체 학생의 학생명, 과목명, 성적을 검색하라(외부조인)

```
SELECT 학생명, 과목명, 성적
FROM   학생 LEFT OUTER JOIN 수강
      ON   학생.학번 = 수강.학번
      LEFT OUTER JOIN 과목
      ON   수강.과목번호 = 과목.과목번호;
```

# SQL: 자체 릴레이션(self)

자체 조인(self join)

담당교수: 김희숙  
(jasmin11@hanmail.net)

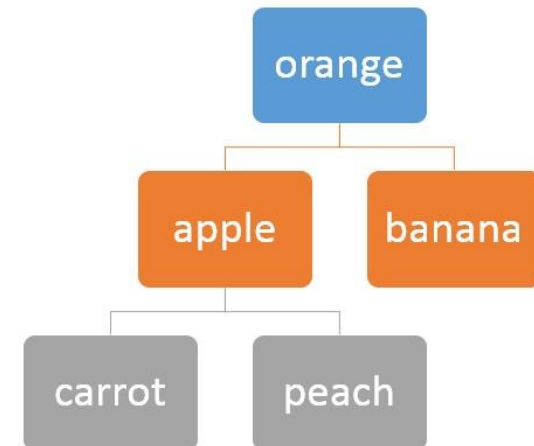
# [요약] 테이블 생성(자체 릴레이션): 추천고객

추천고객

고객 릴레이션의  
기본키

고객아이디	고객이름	나이	등급	직업	적립금	추천고객
apple	김현준	20	gold	학생	1000	orange
banana	정소화	25	vip	간호사	2500	orange
carrot	원유선	28	gold	교사	4500	apple
orange	정지영	22	silver	학생	0	NULL

고객 릴레이션의  
외래키



# [요약] 테이블 1개(자체 릴레이션): emp

❖ 자체 릴레이션: 자신의 테이블을 참조하는 관계(self join)

-- [실습] 사원명과 직속상사명을 검색하라

[예제] 사원명과 직속상사이름을 검색하라

emp 테이블					--실행결과		
	empno	empname	manager	dno		사원명	직속상사명
1	1003	조민희	3011	1	1	조민희	이수민
2	2106	김창섭	3426	2	2	김창섭	박영권
3	3011	이수민	NULL	1	3	박영권	이수민
4	3426	박영권	3011	3	4	최종철	김창섭
5	3427	최종철	2106	3			

--풀이과정

emp E					emp M				
	empno	empname	manager	dno		empno	empname	manager	dno
1	1003	조민희	3011	1	1	1003	조민희	3011	1
2	2106	김창섭	3426	2	2	2106	김창섭	3426	2
3	3011	이수민	NULL	1	3	3011	이수민	NULL	1
4	3426	박영권	3011	3	4	3426	박영권	3011	3
5	3427	최종철	2106	3	5	3427	최종철	2106	3

--SQL 답안을 작성하시오

# [요약] 테이블 1개(자체 릴레이션): emp (ans)

❖ 자체 릴레이션: 자신의 테이블을 참조하는 관계(self join)

-- [실습] 사원명과 직속상사명을 검색하라

[예제] 사원명과 직속상사이름을 검색하라

emp 테이블					--실행결과	
	empno	empname	manager	dno	사원명	직속상사명
1	1003	조민희	3011	1	1	조민희
2	2106	김창섭	3426	2	2	김창섭
3	3011	이수민	NULL	1	3	박영권
4	3426	박영권	3011	3	4	이수민
5	3427	최종철	2106	3	5	김창섭

--풀이과정

emp E					emp M				
	empno	empname	manager	dno		empno	empname	manager	dno
1	1003	조민희	3011	1	1	1003	조민희	3011	1
2	2106	김창섭	3426	2	2	2106	김창섭	3426	2
3	3011	이수민	NULL	1	3	3011	이수민	NULL	1
4	3426	박영권	3011	3	4	3426	박영권	3011	3
5	3427	최종철	2106	3	5	3427	최종철	2106	3

--SQL 답안을 작성하시오



사원명	관리자 이름
조민희	이수민
김창섭	박영권
박영권	이수민
최종철	김창섭



-- [실습] 사원명과 직속상사명을 검색하라

```
select E.empname "사원명", M.empname "직속상사명"
from emp E, emp M
where E.manager = M.empno;
```

# [실습] 테이블 생성(자체 릴레이션)

[실습 self] (테이블 1개) emp 테이블 생성(자체 릴레이션)

(테이블 1개)

emp

외래키

empno	empname	manager	dno
1003	조민희	3011	1
2106	김창섭	3426	2
3011	이수민		1
3426	박영권	3011	3
3427	최종철	2106	3

테이블명	열 이름	데이터 형식	NULL 유무	기본키	외래키	FK 테이블명	FK 열 이름	비고
emp	empno	char(4)	NOT NULL	PK				
	empname	varchar(10)						
	manager	?			?	?	?	
	dno	char(1)						



