

데이터베이스관리

11주차

담당교수: 김희숙
(jasmin11@hanmail.net)

복습

(SQL: Group by)

담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 집계함수

-- 사원(직원코드,성명,직책,연봉)

[실습] 집계함수, 그룹화(GROUP BY)

사원

직원코드	성명	직책	연봉
161353	김미나	대리	2300
181323	최영락	사원	1900
151453	홍진호	과장	2800
135485	구준표	과장	3000
104895	강나래	팀장	3600
165484	김하늘	대리	2400

COUNT(*): 널 값 포함
COUNT(필드): 널 값 제외

SUM()
AVG()
MAX()
MIN()

SELECT
FROM
GROUP BY
HAVING

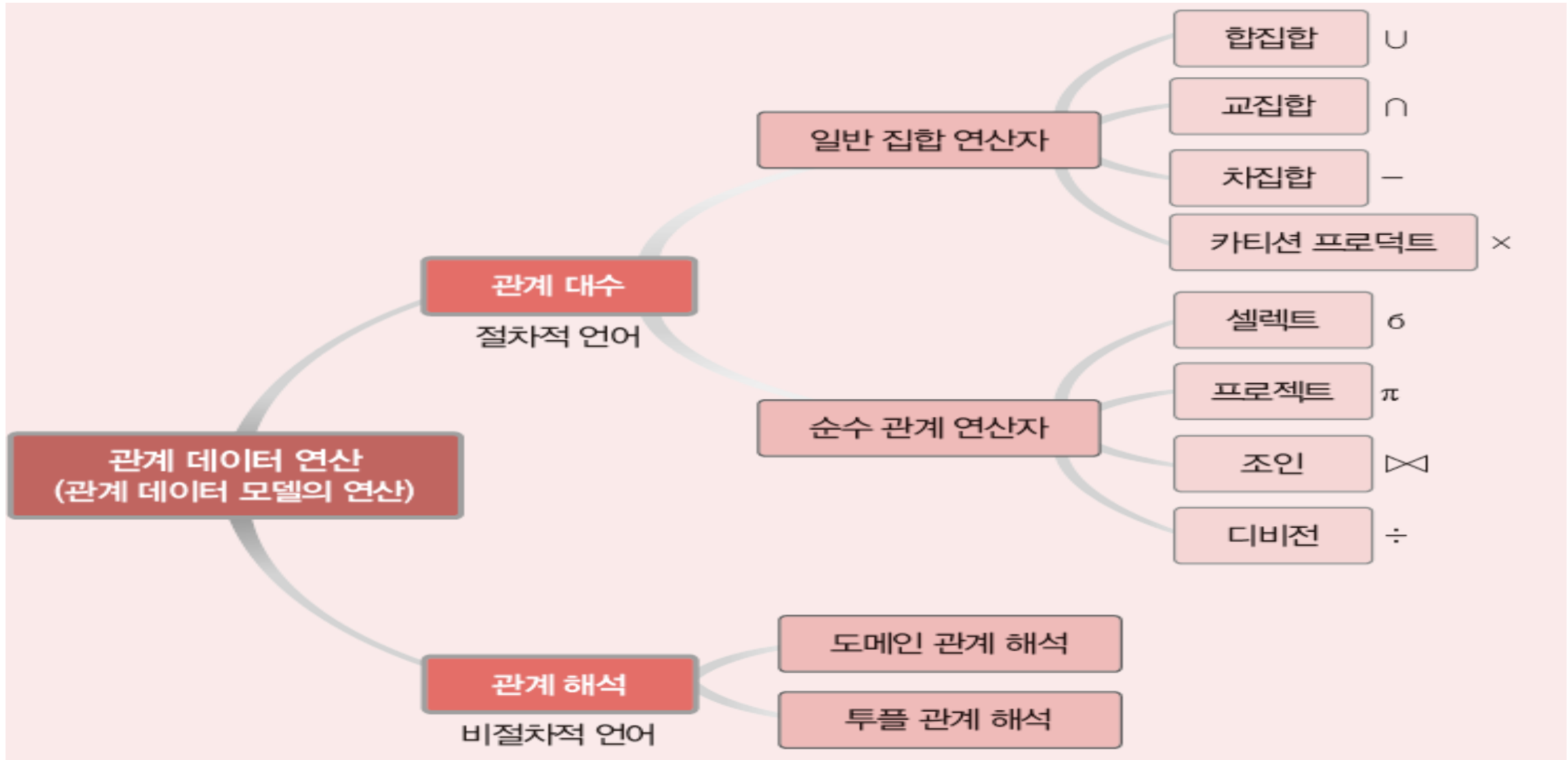
SELECT
FROM
WHERE
GROUP BY
HAVING
ORDER BY

관계대수-01

11주차

담당교수: 김희숙
(jasmin11@hanmail.net)

관계대수



[예제] 합집합, 교집합, 차집합

[예제] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

$$A = \{1, 3, 6\}$$

$$B = \{2, 3, 7\}$$

$$A \cup B = \{1, 2, 3, 6, 7\}$$

$$A \cap B = \{3\}$$

$$A - B = \{1, 6\}$$

* 집합

1. 합집합 UNION
2. 교집합 INTERSECT
3. 차집합 MINUS (Oracle)
EXCEPT (MS SQL)

[실습1-2] (관계대수) 다음을 관계대수로 나타내고 SQL과 실행결과를 작성하라

정회원

	번호	이름	주민번호	휴대폰번호	이메일	등록일
▶	1	홍길동	820416-1234567	(011) 123-1231	gdhong@hitel.net	20070302
	2	임꺽정	830507-2345678	(010) 122-1222	jung@hanmail.net	20050422

준회원

	번호	이름	주민번호	휴대폰번호	이메일	등록일
▶	1	홍길동	820416-1234567	(011) 123-1231	gdhong@hitel.net	20070302
	3	박찬호	850321-1456789	(010) 133-1231	chpark@hanmail.net	20090512
	4	선동열	761122-1889911	(010) 144-1222	sun@naver.com	20080605

-- [Quiz] 정회원과 준회원을 **UNION** 연산자를 사용한 것과, **UNION ALL** 연산자를 사용한 것의 차이점을 확인하라

* 집합

1. 합집합 UNION
2. 교집합 INTERSECT
3. 차집합 MINUS (Oracle)
EXCEPT (MS SQL)

-- 합집합

```
select 번호, 이름 from 정회원
UNION
select 번호, 이름 from 준회원;
```

-- 합집합(중복 포함)

```
select 번호, 이름 from 정회원
UNION ALL
select 번호, 이름 from 준회원;
```

[요약] 관계대수



❖ 관계대수: 수학 수식구조와 거의 동일, 관계대수의 실행결과는 새로운 테이블이 된다

4. 카티션 프로덕트(product)

5. **조인** : 두 개 이상의 테이블에서
원하는 결과를 검색
(공통 필드)

- 1) **내부조인**: 두 개의 릴레이션에서
일치하는 행을 검색
- 2) **외부조인**: 두 개의 릴레이션에서
일치하지 않는 행도 검색

조인 연산의 구분

기본연산 : 세타조인(\bowtie_{θ}), 동등조인(\bowtie), 자연조인(\bowtie_N)

확장된 조인 연산 : 세미조인(\ltimes, \rtimes), 외부조인($\ltimes, \rtimes, \bowtie$)

내부조인

(theta join, equi join, natural join)

외부조인

(left outer join, right outer join, full outer join)

02 관계 대수

❖ 순수 관계 연산자 – 조인

-- 내부조인(inner join)

```
select 고객아이디,고객이름,나이,등급,  
       주문번호,주문제품,수량  
from   고객 INNER JOIN 주문  
on     고객아이디 = 주문고객;
```

-- 내부조인(inner join)

```
select 고객아이디,고객이름,나이,등급,  
       주문번호,주문제품,수량  
from   고객 , 주문  
where  고객아이디 = 주문고객;
```

고객 릴레이션

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

주문 릴레이션

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

↓ 조인 연산

고객 ⋈ 주문

고객아이디	고객이름	나이	등급	주문번호	주문제품	수량
apple	김현준	20	gold	1001	진짜우동	10
banana	정소화	25	vip	1003	그대로만두	11
carrot	원유선	28	gold	1002	맛있는파이	5

그림 6-17 조인 연산의 수행 과정 예 : 고객과 주문 릴레이션

02 관계 대수

❖ 순수 관계 연산자 – 조인

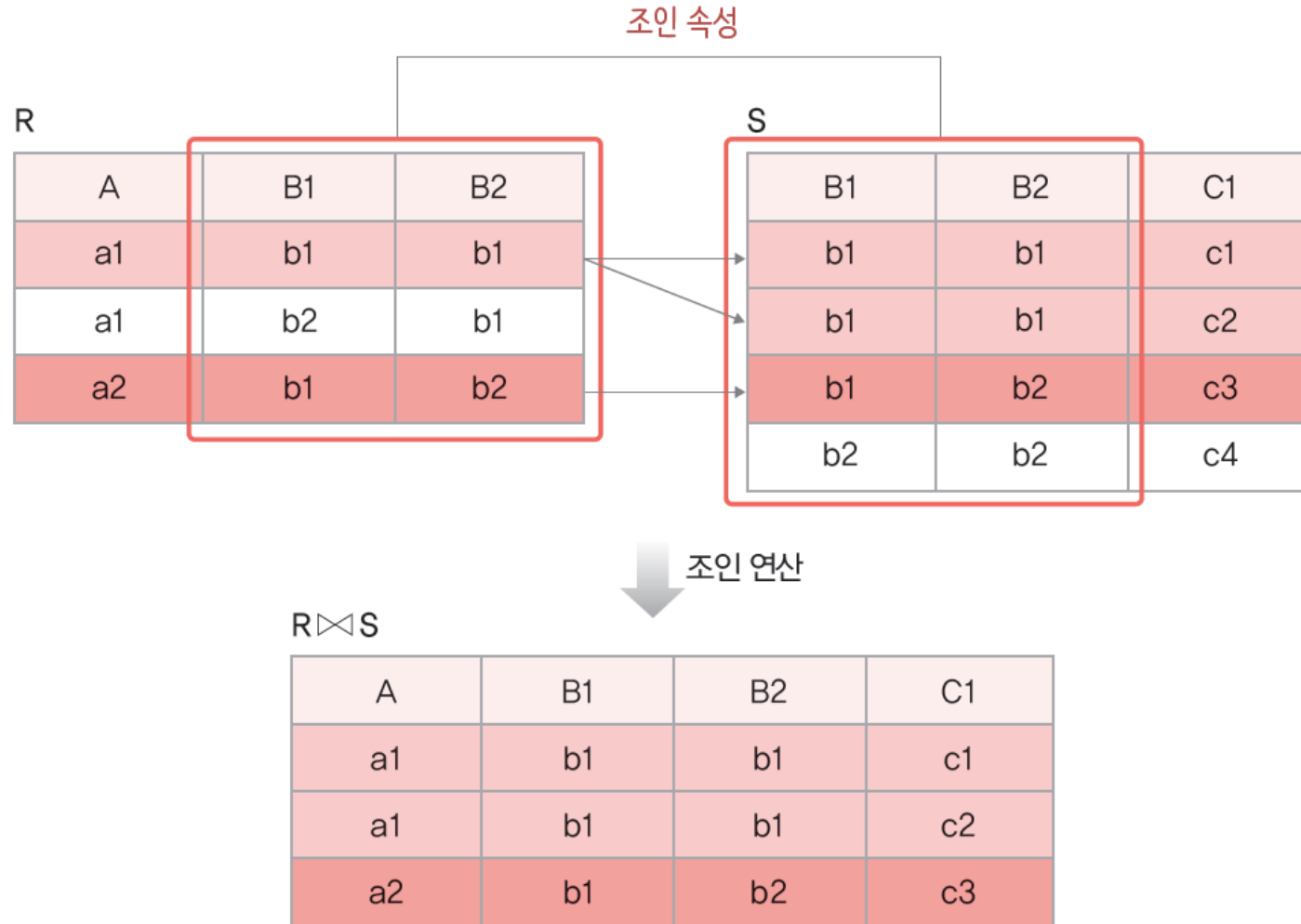


그림 6-18 2개의 속성으로 이루어진 조인 속성을 이용하는 조인 연산의 예 : R과 S 릴레이션

02 관계 대수

- 세타 조인(theta join, θ -join)

- 자연 조인에 비해 더 일반화된 조인
- 주어진 조인 조건을 만족하는 두 릴레이션의 모든 튜플을 연결하여 생성된 새로운 튜플로 결과 릴레이션을 구성
- 결과 릴레이션의 차수는 두 릴레이션의 차수를 더한 것과 같음
- 표현법 : 릴레이션1 $\bowtie_{A\theta B}$ 릴레이션2
 - θ 는 비교 연산자(>, \geq , <, \leq , =, \neq)를 의미

- 동일 조인(equi-join)

- θ 연산자가 “=”인 세타 조인을 의미

02 관계 대수

❖ 동일 조인

-- 내부조인(inner join)

select *

from 고객 INNER JOIN 주문

on 고객아이디 = 주문고객;

고객 릴레이션

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

주문 릴레이션

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

동일 조인 연산

고객 ⋈ 고객아이디=주문고객 주문

고객아이디	고객이름	나이	등급	주문번호	주문고객	주문제품	수량
apple	김현준	20	gold	1001	apple	진짜우동	10
banana	정소화	25	vip	1003	banana	그대로만두	11
carrot	원유선	28	gold	1002	carrot	맛있는파이	5

그림 6-19 동일 조인 연산의 예: 고객과 주문 릴레이션

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

(담당교수: 김희숙)

[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
	임꺽정	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
	임꺽정	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
▶	임꺽정	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 ⋈_N 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)

-- 고객(고객아이디, 고객이름, 나이, 등급)

-- 주문(주문번호, 주문고객, 주문제품, 수량)

```
CREATE TABLE 고객 (  
    고객아이디 VARCHAR(20)      NOT NULL,  
    고객이름   VARCHAR(10),  
    나이       INT,  
    등급       VARCHAR(10)      NOT NULL,  
    PRIMARY KEY(고객아이디)  
);  
  
CREATE TABLE 주문 (  
    주문번호   CHAR(4) NOT NULL,  
    주문고객   VARCHAR(20),  
    주문제품   VARCHAR(20),  
    수량       INT,  
    PRIMARY KEY(주문번호),  
    FOREIGN KEY(주문고객) REFERENCES 고객(고객아이디)  
);
```

-- 고객(고객아이디, 고객이름, 나이, 등급)

-- 주문(주문번호, 주문고객, 주문제품, 수량)

-- 고객(고객아이디, 고객이름, 나이, 등급)

```
INSERT INTO 고객 VALUES ('apple', '김현준', 20, 'gold');
```

```
INSERT INTO 고객 VALUES ('banana', '정소화', 25, 'vip');
```

```
INSERT INTO 고객 VALUES ('carrot', '원유선', 28, 'gold');
```

```
INSERT INTO 고객 VALUES ('orange', '정지영', 22, 'silver');
```

-- 주문(주문번호, 주문고객, 주문제품, 수량)

```
INSERT INTO 주문 VALUES ('1001', 'apple', '진짜우동', 10);
```

```
INSERT INTO 주문 VALUES ('1002', 'carrot', '맛있는파이', 5);
```

```
INSERT INTO 주문 VALUES ('1003', 'banana', '그대로만두', 45);
```

```
select * from 고객;
```

```
select * from 주문;
```

[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	<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 ⋈ _N S1																						
4. R1 ⋈ _{A >= C} 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 ⋈ _N S1																						
4. R1 ⋈ _{A ≥ C} 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 ⋈_N S1

4. R1 ⋈_{A >= C} S1

-- 관계 연산자

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

-- 4. 세타조인(theta join)

select *

from R1, S1

where R1.A >= C;



[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

1. 카티션 프로덕트(product)
2. 동등조인(equi join)
3. 자연조인(natural join)
4. 왼쪽외부조인(left outer join)
5. 오른쪽외부조인(right outer join)
6. 완전외부조인(full outer join)

-- 1. 카티션 프로덕트

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

[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

- 1. 카티션 프로덕트(product)
- 2. 동등조인(equi join)
- 3. 자연조인(natural join)
- 4. 왼쪽외부조인(left outer join)
- 5. 오른쪽외부조인(right outer join)
- 6. 완전외부조인(full outer join)

-- 1. 카티션 프로덕트

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

-- 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;
```

[예제] [Quiz]

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

관계대수 / SQL																			
freshman		dmember																	
<table><tr><th>name</th><th>address</th></tr><tr><td>김광식</td><td>서울</td></tr><tr><td>김현정</td><td>대전</td></tr><tr><td>조영수</td><td>대전</td></tr></table>	name	address	김광식	서울	김현정	대전	조영수	대전		<table><tr><th>name</th><th>dept_name</th></tr><tr><td>김광식</td><td>컴퓨터공학과</td></tr><tr><td>김현정</td><td>산업공학과</td></tr><tr><td>이진영</td><td>전자공학과</td></tr></table>	name	dept_name	김광식	컴퓨터공학과	김현정	산업공학과	이진영	전자공학과	
name	address																		
김광식	서울																		
김현정	대전																		
조영수	대전																		
name	dept_name																		
김광식	컴퓨터공학과																		
김현정	산업공학과																		
이진영	전자공학과																		
1-1) 관계대수		1. 카티션 프로덕트(product)																	
1-2) SQL		2. 동등조인(equi join)																	
1-3) 실행결과		3. 자연조인(natural join)																	
		4. 왼쪽외부조인(left outer join)																	
		5. 오른쪽외부조인(right outer join)																	
		6. 완전외부조인(full outer join)																	

-- 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		dmember																	
<table><tr><th>name</th><th>address</th></tr><tr><td>김광식</td><td>서울</td></tr><tr><td>김현정</td><td>대전</td></tr><tr><td>조영수</td><td>대전</td></tr></table>	name	address	김광식	서울	김현정	대전	조영수	대전		<table><tr><th>name</th><th>dept_name</th></tr><tr><td>김광식</td><td>컴퓨터공학과</td></tr><tr><td>김현정</td><td>산업공학과</td></tr><tr><td>이진영</td><td>전자공학과</td></tr></table>	name	dept_name	김광식	컴퓨터공학과	김현정	산업공학과	이진영	전자공학과	
name	address																		
김광식	서울																		
김현정	대전																		
조영수	대전																		
name	dept_name																		
김광식	컴퓨터공학과																		
김현정	산업공학과																		
이진영	전자공학과																		
1-1) 관계대수		1. 카티션 프로덕트(product)																	
1-2) SQL		2. 동등조인(equi join)																	
1-3) 실행결과		3. 자연조인(natural join)																	
		4. 왼쪽외부조인(left outer join)																	
		5. 오른쪽외부조인(right outer join)																	
		6. 완전외부조인(full outer join)																	

-- 4. 왼쪽 외부조인(left outer join)

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


[예제] [Quiz]

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

관계대수 / SQL																			
freshman		dmember																	
<table><tr><th>name</th><th>address</th></tr><tr><td>김광식</td><td>서울</td></tr><tr><td>김현정</td><td>대전</td></tr><tr><td>조영수</td><td>대전</td></tr></table>	name	address	김광식	서울	김현정	대전	조영수	대전		<table><tr><th>name</th><th>dept_name</th></tr><tr><td>김광식</td><td>컴퓨터공학과</td></tr><tr><td>김현정</td><td>산업공학과</td></tr><tr><td>이진영</td><td>전자공학과</td></tr></table>	name	dept_name	김광식	컴퓨터공학과	김현정	산업공학과	이진영	전자공학과	
name	address																		
김광식	서울																		
김현정	대전																		
조영수	대전																		
name	dept_name																		
김광식	컴퓨터공학과																		
김현정	산업공학과																		
이진영	전자공학과																		
1-1) 관계대수		1. 카티션 프로덕트(product)																	
1-2) SQL		2. 동등조인(equi join)																	
1-3) 실행결과		3. 자연조인(natural join)																	
		4. 왼쪽외부조인(left outer join)																	
		5. 오른쪽외부조인(right outer join)																	
		6. 완전외부조인(full outer join)																	

-- 5. 오른쪽 외부조인(right outer join)

```
select freshman.name, address, dept_name
from  freshman RIGHT OUTER 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

- 1. 카티션 프로덕트(product)
- 2. 동등조인(equi join)
- 3. 자연조인(natural join)
- 4. 왼쪽외부조인(left outer join)
- 5. 오른쪽외부조인(right outer join)
- 6. 완전외부조인(full outer join)

-- 6. 완전 외부조인(full outer join)

[Quiz]

11주차

담당교수: 김희숙
(jasmin11@hanmail.net)

[Quiz] (MySQL)

(MySQL)

테이블 생성

SELECT문제

(화면캡처 본문작성)

□ [Quiz] **MySQL 프로그램 사용**하여 테이블 작성하고
SELECT 문제 만들고 실습하시오

1) 제출방법: 네이버카페-게시판제출: 과제

게시글에 **답글**로 작성하고 **게시글 본문**에 화면캡처 첨부

2) 테이블 생성

SELECT 문제, SQL 답안, 실행결과 화면캡처

* (작성한 **SQL**파일은 저장해 두세요: 각자 별도로 보관해 두시기 바랍니다)

[실습] [Quiz 2] (MySQL)

[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	바경오권	과장	4377	3000000	1
3427	최종철	사원	3011	1500000	3
4377	이성래	이사	NULL	5000000	2

- 1) 부서의 모든 필드를 검색하라
- 2) 부서의 부서번호와 부서이름을 검색하라 (deptno는 부서번호, deptname은 부서이름 으로 검색)
- 3) 사원의 직급을 검색하라
- 4) 사원의 직급을 한번씩만 검색하라 (중복제거) (DISTINCT)
- 5) 2번 부서에 근무하는 직원들에 관한 모든 정보를 검색하라
- 6) 이씨 성을 가진 직원들의 이름, 직급, 부서번호를 검색하라
- 7) 직급이 과장이면서 1번 부서에서 근무하는 직원들의 이름과 급여를 검색하라
- 8) 직급이 과장이면서 1번 부서에 속하지 않은 직원들의 이름과 급여를 검색하라
- 9) 급여가 3000000원 이상이고, 4500000원 이하인 직원의 이름, 직급, 급여를 검색하라
- 9-1) BETWEEN ... AND 사용
- 9-2) 부등호 사용

[실습] [Quiz 2] (MySQL)

[실습]

-- department(deptno, deptname, floor)

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

department

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

employee

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	바경오권	과장	4377	3000000	1
3427	최종철	사원	3011	1500000	3
4377	이성래	이사	NULL	5000000	2

- 10) 1번 부서나 3번 부서에 소속된 직원들에 관한 모든 정보를 검색하라 (IN 사용)
- 11) 직급이 과장인 직원들에 대하여 이름, 현재급여, 급여가 10% 인상됐을 때의 값을 검색하라
- 12) 2번 부서에 근무하는 직원들의 급여, 직급, 이름을 검색하여 급여의 오름차순으로 정렬하라
- 13) 직원들의 평균 급여와 최대급여를 검색하라
- 14) 직원들이 속한 부서번호별로 그룹화하고 부서번호, 평균급여, 최대급여를 검색하라
- 15) 직원들이 속한 부서번호별로 그룹화하고, 평균급여가 2500000원 이상인
- 부서에 대해서 부서번호, 평균급여, 최대급여를 검색하라

[실습] [Quiz 2] (MySQL)

[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	바경오 권	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)

-- 18) 사원에 대해서 사원이름과 직속상사의 이름을 검색하라(Self-Join) : (해당 문제는 제외)

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

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

실습(Quiz)

담당교수: 김희숙
(jasmin11@hanmail.net)

[실습] [Quiz 1] fruit 테이블(MySQL)

[Quiz 1]

1) 다음과 같은 fruit 테이블을 생성하시오
(기본키: 과일번호)

데이터 입력하는 SQL문법을 작성하라

fruit

과일번호	과일이름	가격	색상	원산지
1	사과	2000	빨간색	한국
2	배	5000	노란색	한국
3	바나나	1500	노란색	인도네시아
4	키위	4000	초록색	호주
5	수박	6000	초록색	한국

- 2-1) 노란색 과일의 이름, 색상을 검색하라.
- 2-2) 과일이름을 오름차순으로 정렬하여 모든 필드를 검색하라
- 2-3) 가격이 5000 이하인 빨간색 과일의 이름을 검색하라
- 2-4) 가격이 2000 ~ 5000 사이의 과일이름과 가격, 원산지를 검색하라(BETWEEN AND)
- 2-5) 색상이 노란색이거나 초록색인 과일이름, 색상을 검색하라 (IN 사용)
- 2-6) 색상이 노란색이거나 초록색인 과일이름, 색상을 검색하라 (논리연산자 OR 사용)
- 2-7) 원산지가 한국이 아닌 과일이름, 원산지를 검색하라

- 3-1) UPDATE 문을 이용하여 사과의 가격을 2배로 하여 수정하라
- 3-2) DELETE 문을 이용하여 원산지가 호주인 과일의 목록을 삭제하라
- 3-3) ALTER TABLE 문을 이용하여 과일정보 필드를 추가하라 (데이터형식 varchar(20))

[실습] [Quiz 2] 고객 테이블(MySQL) (stu)

고객

고객아이디	고객이름	나이	등급	직업	적립금
apple	정소화	20	gold	학생	1000
banana	김선우	25	vip	간호사	2500
carrot	고명석	28	gold	교사	4500
orange	김용축	22	silver	학생	NULL
melon	성원용	35	gold	회사원	5000
peach	오형준	NULL	silver	의사	300
pear	채광주	31	silver	회사원	500

- 1) 직업이 학생, 간호사, 교사인 고객의 이름, 직업, 등급을 검색
- 2) 적립금이 300 이상인 고객 중에서 나이를 알 수 없는 고객아이디, 나이, 적립금을 검색하라
- 3) 등급이 gold인 고객 중에서 고객아이디, 등급, 적립금을 검색하라. 단, 등급을 오름차순 정렬하고 등급이 같으면 적립금의 내림차순 정렬하라
- 4) 등급별 적립금 평균을 검색하라,
(단, 적립금 평균이 1000원 이상인 등급에 대해 검색)

- 고객 테이블은
- 고객아이디, 고객이름, 나이, 등급, 직업, 적립금 속성으로 구성되고,
- 고객아이디 속성이 기본키다.
- 고객이름과 등급 속성은 값을 반드시 입력해야 하고,
- 적립금 속성은
- 값을 입력하지 않으면 0이 기본으로 입력되도록 한다