

데이터베이스

인공지능소프트웨어학과

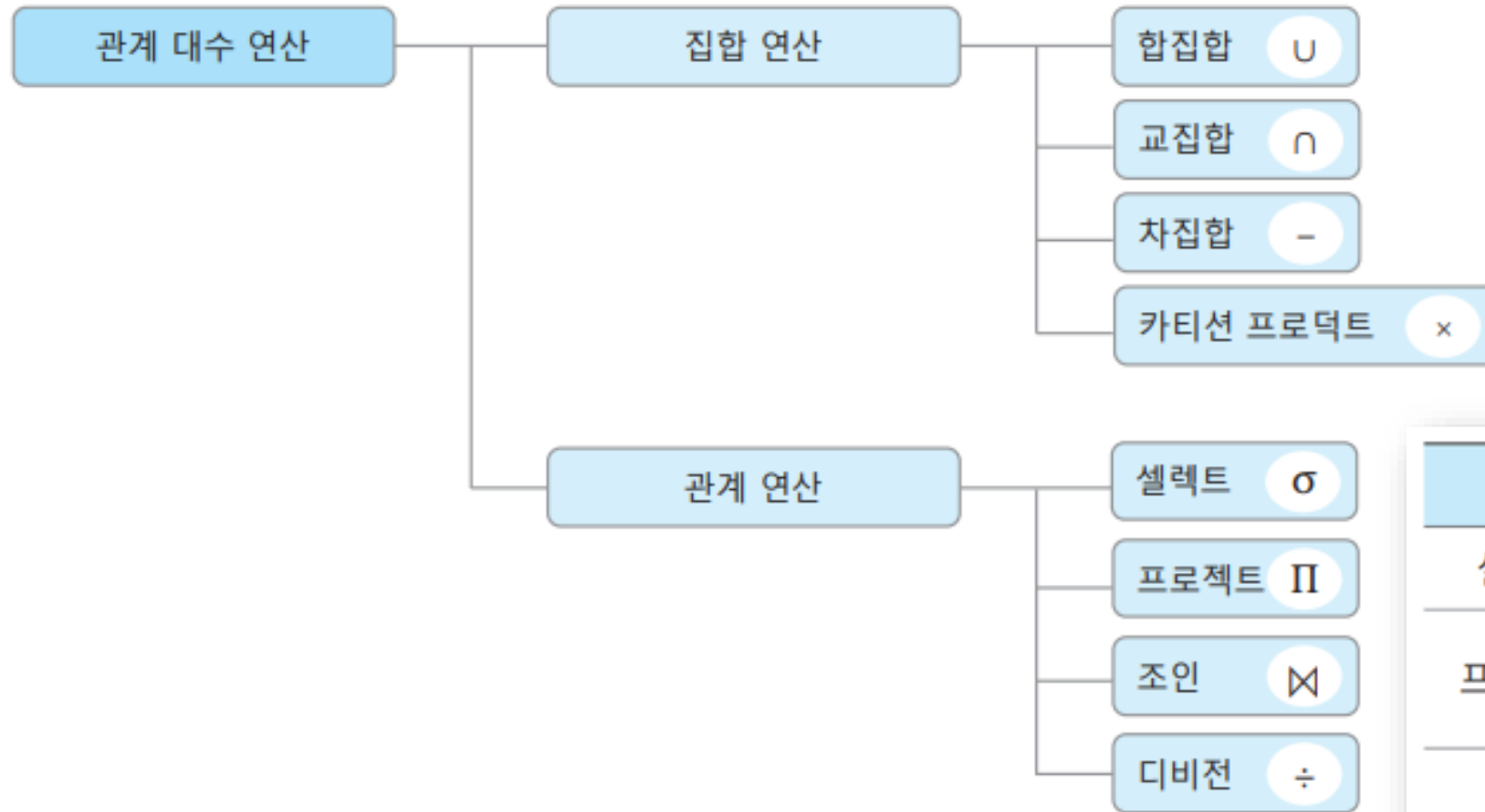
담당교수: 김희숙
(jasmin11@hanmail.net)

데이터베이스

12주차

담당교수: 김희숙
(jasmin11@hanmail.net)

관계대수



연산	연산자	형식
선택	σ	$\sigma_{\text{선택_조건식}}(R)$
프로젝트	Π	$\Pi_{\text{속성_리스트}}(R)$
조인	\bowtie	$R1 \bowtie_{\text{조인_조건식}} R2$
디비전	\div	$R1 \div R2$

[요약] 관계대수



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

1. 합집합, 교집합, 차집합 $\cup \cap -$

2. 실렉션(selection) σ

3. 프로젝션(projection) π

4. 카티션 프로덕트 \times

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

* 집합

1) 합집합 **UNION**

2) 교집합 **INTERSECT**

3) 차집합 **MINUS** (Oracle)
EXCEPT (MS SQL)

조인 연산의 구분

기본연산 : 세타조인(\bowtie_θ), 동등조인(\bowtie), 자연조인(\bowtie_N)

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

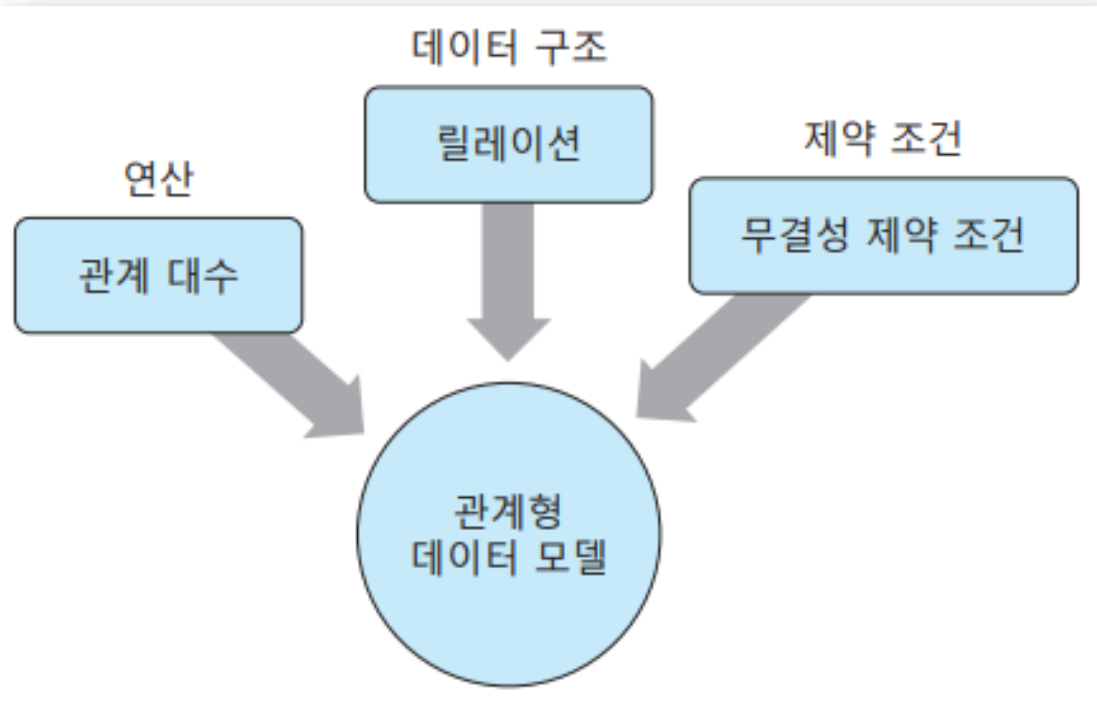
내부조인

(theta join, equi join, natural join)

외부조인

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

[요약] 데이터 모델 (S, O, C)



* 데이터 모델

- 1) 데이터 구조 **Structure**
- 2) 연산 **Operation**
- 3) 제약조건 **Constraint**

데이터 모델 = 데이터 구조 + 연산 + 제약조건

[복습] 조인(join)

(내부조인, 외부조인)

담당교수: 김희숙
(jasmin11@hanmail.net)

[예제] [복습]



[복습] (조인) 다음을 관계대수로 나타내고 SQL 문법과 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

[예제] [복습]

노트
풀이

[복습] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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

1-1) 관계대수

1-2) SQL

1-3) 실행결과

-- 1. 카티션 프로덕트

-- 2. 동등조인

-- 3. 자연조인

[예제] [복습]

노트
풀이

[복습] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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] [복습]

[복습] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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] [복습]

[복습] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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)

(지원하는 문법 없음)

[Quiz] 관계대수 / 조인

(관계대수식 / SQL문법 / 실행결과)

담당교수: 김희숙
(jasmin11@hanmail.net)

[예제] [Quiz]

[Quiz 1-1] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 실행결과를 작성하라

노트
풀이

R1	A	B	S1	A	C	D	다음 결과는? 1. R1 X S1 2. R1 ⋈ S1 3. R1 ⋈ _N S1 4. R1 ⋈ _{A ≥ C} S1
	1	4		1	3	4	
	2	5		2	2	6	
				3	1	9	

-- **노트**에 각 실행결과를 작성하시오

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

[예제] [Quiz]

[Quiz 1-1] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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

-- 1. 카티션 프로덕트

select *

from R1, S1;

-- 2. 동등조인(equi join)

select *

from R1, S1

where R1.A = S1.A;

[예제] [Quiz]

[Quiz 1-1] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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 1-1] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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;

[예제] [Quiz]

[Quiz 1-2] (조인) 다음을 관계대수로 나타내고 SQL 문법과 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 1-3] (조인) 다음을 관계대수로 나타내고 SQL 문법과 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 프로덕트
- 2. 동등조인
- 3. 자연조인
- 4. 왼쪽외부조인
- 5. 오른쪽외부조인
- 6. 완전외부조인

[예제] [Quiz]



[Quiz 1-4] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 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;
```

[실습] 합집합



[실습 1-1]

```
-- 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 1-2]

```
-- 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] [관계대수] (MySQL)

[실습 2-1] 관계대수, 조인 실행결과

[Quiz 1] 다음을 관계대수로 나타내고 SQL 과 실행결과를 작성하라

R1	R2	*집합
A	A	
B	B	
C	C	
a1	a1	1. 합집합 UNION
a2	a2	2. 교집합 INTERSECT
a3	a3	3. 차집합 MINUS (Oracle)
		EXCEPT (MS SQL)

주요 연산자	사용 예	결과												
선택선 (σ)	$\sigma_{A=a1 \text{ or } A=a2} (R1)$													
프로젝션 (π)	$\pi_{A,B} (R2)$													
합집합 (∪)	$R1 \cup R2$													
차집합 (-)	$R1 - R2$													
조인 (⋈)	$R1 \bowtie_{R1.C=R2.C} R2$	<table><tr><th>R1.A</th><th>R1.B</th><th>R1.C</th><th>R2.A</th><th>R2.B</th><th>R2.C</th></tr><tr><td></td><td></td><td></td><td></td><td></td><td></td></tr></table>	R1.A	R1.B	R1.C	R2.A	R2.B	R2.C						
R1.A	R1.B	R1.C	R2.A	R2.B	R2.C									

[실습] [관계대수] (MySQL)

[실습 2-2] 관계대수, 조인 실행결과

[Quiz 2] 다음을 관계대수로 나타내고 SQL 과 실행결과를 작성하라

질의 2-1 마당서점에서 판매하는 도서 중 8,000원 이하인 도서를 검색하시오.

$\sigma_{\text{가격} \leq 8000}$ (도서)

질의 2-2 고객 릴레이션에서 이름, 주소, 핸드폰을 검색하시오.

$\pi_{\text{이름, 주소, 핸드폰}}$ (고객)

질의 2-7 고객과 고객의 주문 사항을 모두 보이시오.

고객 $\bowtie_{\text{고객.고객번호=주문.고객번호}}$ 주문 (동등조인)

질의 2-8 고객과 고객의 주문 사항을 모두 보이시오.

고객 $\bowtie_{N(\text{고객.고객번호=주문.고객번호})}$ 주문 (자연조인)

질의 2-11 마당서점의 도서 중 가격이 8,000원 이하인 도서이름과 출판사를 보이시오.

$\pi_{\text{도서이름, 출판사}} (\sigma_{\text{가격} \leq 8000} \text{ 도서})$

[실습] [관계대수] (MySQL)

[실습 2-3] 관계대수, 조인 실행결과

[Quiz 3] 박지성의 주문번호, 이름, 판매가격을 검색하라

$\pi_{\text{주문.주문번호, 고객.이름, 주문.판매가격}} (\sigma_{\text{고객.고객번호=주문.고객번호 AND 고객.이름='박지성'}} (\text{고객} \times \text{주문}))$

$\pi_{\text{주문번호, 이름, 판매가격}} (\sigma_{\text{이름='박지성'}} (\text{고객} \bowtie_{\text{고객.고객번호=주문.고객번호}} \text{주문}))$

관계대수-01

(집합)

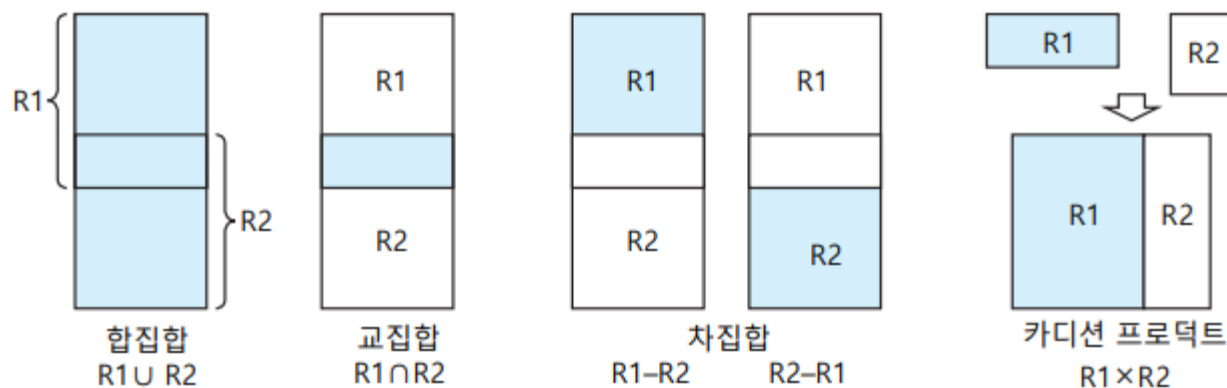
담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 관계 대수

- 일반 집합 연산자(set operation)

- 릴레이션을 튜플 집합 또는 속성 집합으로 간주하여 이를 처리하는 연산 그룹

연산	연산자	형식	설명
합집합	\cup	$R1 \cup R2$	릴레이션 R1과 R2의 모든 튜플을 반환
교집합	\cap	$R1 \cap R2$	릴레이션 R1과 R2의 중복 튜플만을 반환
차집합	$-$	$R1 - R2$	릴레이션 R1에만 속하고 R2에는 속하지 않는 튜플만을 반환
카티션 프로덕트	\times	$R1 \times R2$	릴레이션 R1과 R2의 각 튜플들을 서로 연결하여 만들어진 모든 튜플 조합(곱)을 반환



[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)

[실습 3-1] (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

* 집합

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

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

-- 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] 관계 대수

- 일반 집합 연산자의 특성
 - 피연산자 2개 필요
 - 릴레이션 2개를 대상으로 연산을 수행
 - 합집합, 교집합, 차집합은 피연산자인 두 릴레이션이 합병 가능해야 한다
 - 합병 가능(**union-compatible**)

[요약] 관계 대수

* 합집합 UNION

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

U

학생2

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s004	이영애	2	여



학생1 U 학생2

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남
s004	이영애	2	여

* 합집합 UNION

2.

3. 차집합 MINUS (Oracle)
EXCEPT (MS SQL)

* 교집합 INTERSECT

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

∩

학생2

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s004	이영애	2	여



학생1 ∩ 학생2

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남

[요약] 관계 대수

* 차집합 MINUS (Oracle) EXCEPT (MS SQL)

학생1	학번	이름	학년	성별
	s001	김연아	4	여
	s002	홍길동	1	남
	s003	이승엽	3	남

-

학생2	학번	이름	학년	성별
	s001	김연아	4	여
	s002	홍길동	1	남
	s004	이영애	2	여

↓

학생1 - 학생2

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남

* 카티션 프로덕트 PRODUCT

학생1	학번	이름	학년	성별
	s001	김연아	4	여
	s002	홍길동	1	남
	s003	이승엽	3	남

×

과목1	과목번호	과목이름	강의실
	c001	정보보호	126
	c002	데이터베이스	137

↓

학생1 × 과목1

학번	이름	학년	성별	과목번호	과목이름	강의실
s001	김연아	4	여	c001	정보보호	126
s001	김연아	4	여	c002	데이터베이스	137
s002	홍길동	1	남	c001	정보보호	126
s002	홍길동	1	남	c002	데이터베이스	137
s003	이승엽	3	남	c001	정보보호	126
s003	이승엽	3	남	c002	데이터베이스	137

[실습] 관계대수(Oracle)

<사원>

사번	이름
1	고길동
2	이순신

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

<직원>

사번	이름
2	이순신
3	둘리

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

```
-- (Oracle)
select * from 사원
INTERSECT
select * from 직원;
```

```
-- (Oracle)
select * from 사원
MINUS
select * from 직원;
```

```
-- (MS SQL)
select * from 사원
EXCEPT
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');
```


관계대수-02

(실렉션, 프로젝션)

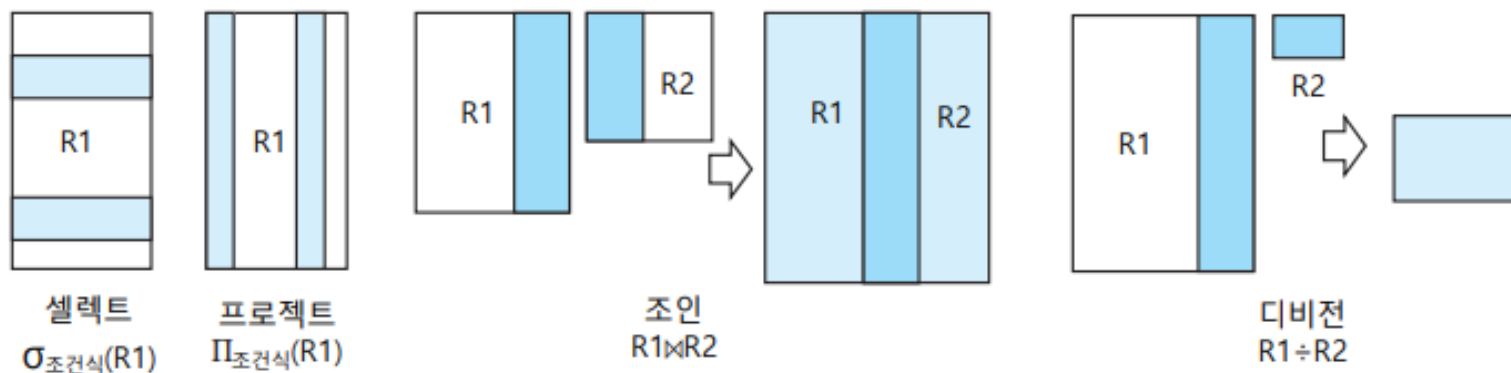
담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 관계 대수

• 순수 관계 연산자(set operation)

- 릴레이션의 구조와 특성을 이용하는 연산자

연산	연산자	형식	설명
선택	σ	$\sigma_{\text{선택_조건식}}(R)$	릴레이션 R에서 선택_조건식을 만족하는 튜플만을 선택하여 반환
프로젝트	Π	$\Pi_{\text{속성_리스트}}(R)$	릴레이션 R에서 속성_리스트에 나열된 속성 값으로만 구성된 튜플을 반환
조인	\bowtie	$R1 \bowtie_{\text{조인_조건식}} R2$	릴레이션 R1과 R2 튜플을 연결한 조합 중에서 조인_조건식을 만족하는 튜플만을 선택하여 반환
디비전	\div	$R1 \div R2$	릴레이션 R2의 모든 튜플과 관련 있는 릴레이션 R1의 튜플만을 반환



[요약] 관계 대수

- **순수 관계 연산자(set operation)**
 - 릴레이션의 구조와 특성을 이용하는 연산자

예) $\sigma_{\text{중간성적} < \text{기말성적}}(\text{성적})$

$\sigma_{\text{이름} = \text{'홍길동'}}(\text{성적})$

$\sigma_{(\text{중간성적} < \text{기말성적}) \wedge (\text{이름} = \text{'홍길동'})}(\text{성적})$

$\sigma_{\text{학년} = 4}(\text{성적})$

예) $\Pi_{\text{이름}, \text{성별}, \text{학년}}(\text{학생1})$

[예제] 실렉션(selection)

[예제] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 실행결과를 작성하라

[예제] (관계대수) 고객 릴레이션에서 등급이 gold인 튜플을 검색하라

1. 관계대수식

$\sigma_{\text{등급}='gold'}(\text{고객})$

2. SQL 문법

```
SELECT *  
FROM 고객  
WHERE 등급='gold';
```

3. 실행결과

고객

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



고객아이디	고객이름	나이	등급	직업	적립금
apple	김현준	20	gold	학생	1000
carrot	원유선	28	gold	교사	4500

[예제] 프로젝션(projection)

[예제] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 실행결과를 작성하라

[예제] (관계대수) 고객 릴레이션에서 고객이름, 등급, 적립금을 검색하라

1. 관계대수식

$\pi_{\text{고객이름, 등급, 적립금}}(\text{고객})$

2. SQL 문법

```
SELECT 고객이름, 등급, 적립금  
FROM 고객
```

3. 실행결과

고객

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



고객이름	등급	적립금
정소화	gold	1000
김선우	vip	2500
고명석	gold	4500
김용욱	silver	0

[예제] 실렉션(selection)

[예제] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 실행결과를 작성하라

학생3

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남
s004	이영애	2	여
s005	송윤아	4	여



학번	이름	학년	성별
s002	홍길동	1	남
s003	이승엽	3	남

$\sigma_{\text{성별}='남'}(\text{학생3})$

```
SELECT *  
FROM 학생3  
WHERE 성별='남';
```

* 관계대수: 관계 연산

- 1. 실렉션(selection) σ
- 2. 프로젝션(projection) Π
- 3. 조인(join) \bowtie

자연조인(natural join) \bowtie_N

[예제] 프로젝션(projection)

[예제] (관계대수) 다음을 관계대수로 나타내고 SQL 문법과 실행결과를 작성하라

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

작성하라

이름	성별	학년
김연아	여	4
홍길동	남	1
이승엽	남	3

* 관계대수: 관계 연산

1. 실렉션(selection) σ
2. **프로젝션(projection)** Π
3. 조인(join) \bowtie
- 자연조인(natural join) \bowtie_N

$\Pi_{\text{이름,성별,학년}}(\text{학생1})$

```
SELECT 이름, 성별, 학년
FROM 학생1;
```

[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 >= 80;
```


[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: [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	π 학년, 학과(학생)	5	DIFFERENCE
6	CARTESIAN	7	σ 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

관계대수-03

(카티션 프로덕트, 조인)

담당교수: 김희숙
(jasmin11@hanmail.net)

[요약] 관계 대수

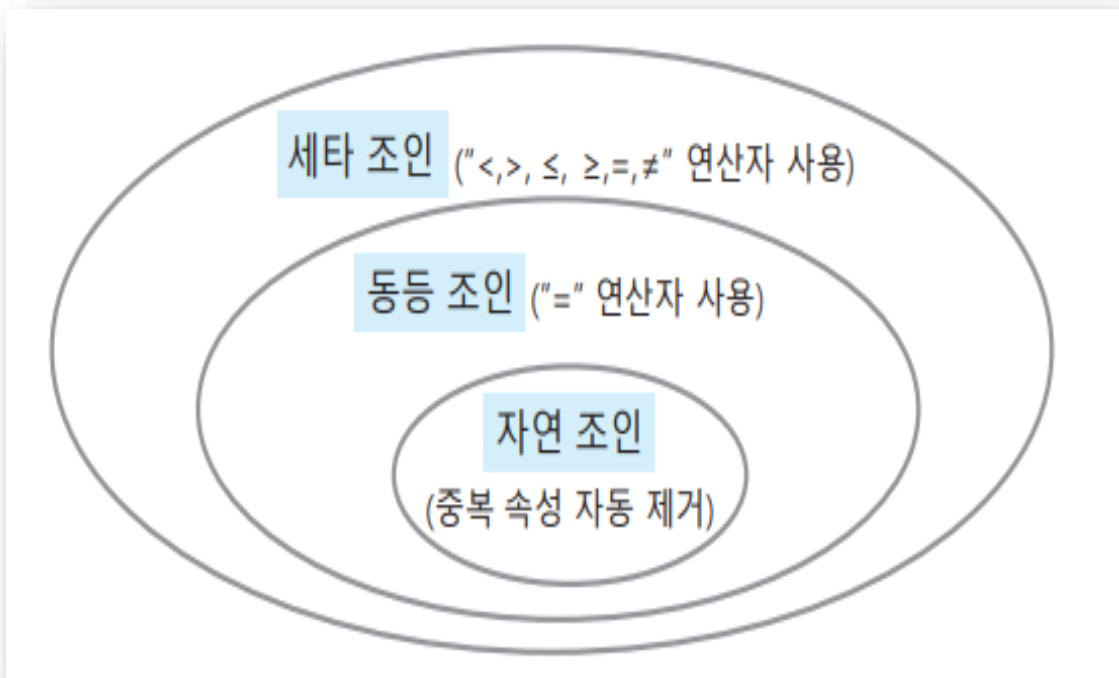
- 순수 관계 연산자 – 조인(join)

- 조인 속성을 이용해 두 릴레이션을 조합하여 결과 릴레이션을 구성
 - 조인 속성의 값이 같은 튜플만 연결하여 생성된 튜플을 결과 릴레이션에 포함
 - 조인 속성 : 두 릴레이션이 공통으로 가지고 있는 속성

- 표현법 : 릴레이션1 \bowtie 릴레이션2

- 자연 조인(natural join)이라고도 한다

- 표현법 : 릴레이션1 \bowtie_N 릴레이션2



[요약] 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;
```

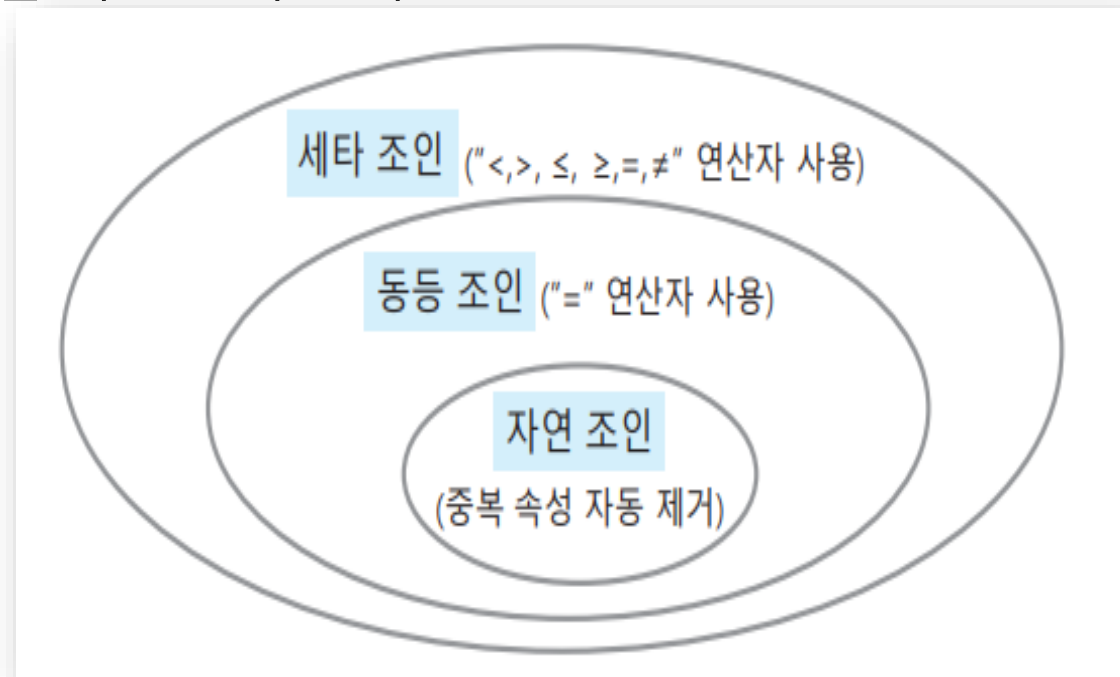
[요약] 관계 대수

- 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 같다
- 표현법 : 릴레이션1 $\bowtie_{A\theta B}$ 릴레이션2
 - θ 는 비교 연산자(>, ≥, <, ≤, =, ≠)를 의미

- 동등 조인(equi-join)

- θ 연산자가 “=”인 세타 조인을 의미



[요약] 관계 대수

❖ 동등 조인 (equi join)

-- 내부조인(inner join)

select *

from 학생1 INNER JOIN 수강2

on 학생1.학번 = 수강2.학번;

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

수강2

학번	과목번호	평가학점
s001	c002	A
s003	c002	B
s002	c001	F
s001	c004	B
s001	c005	C

학생1 \bowtie 학생1.학번=수강2.학번 수강2



학번	이름	학년	성별	학번	과목번호	평가학점
s001	김연아	4	여	s001	c002	A
s001	김연아	4	여	s001	c004	B
s001	김연아	4	여	s001	c005	C
s002	홍길동	1	남	s002	c001	F
s003	이승엽	3	남	s003	c002	B

[요약] 관계 대수

❖ 자연 조인 (natural join) 학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

수강2

학번	과목번호	평가학점
s001	c002	A
s003	c002	B
s002	c001	F
s001	c004	B
s001	c005	C

-- 내부조인(inner join)

```
select 학생1.학번,이름,학년,성별,과목번호,평가학점
from 학생1 INNER JOIN 수강2
on 학생1.학번 = 수강2.학번;
```

학생1 ⋈_{N(학생1.학번,수강2.학번)} 수강2



학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s001	김연아	4	여	c005	C
s002	홍길동	1	남	c001	F
s003	이승엽	3	남	c002	B

[요약] 관계 대수

❖ 외부 조인 (왼쪽 외부 조인)

-- 왼쪽 외부 조인(left outer join)

select *

from 학생1 LEFT OUTER JOIN 수강4

on 학생1.학번 = 수강4.학번;

완전 외부 조인(\bowtie)

왼쪽 외부 조인(\bowtie)

오른쪽 외부 조인(\bowtie)

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

수강4

학번	과목번호	평가학점
s001	c002	A
s002	c001	F
s001	c004	B
s004	c005	C

학생1 \bowtie (학생1.학번,수강4.학번) 수강4



학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F
s003	이승엽	3	남	NULL	NULL

학생1 \bowtie $N_{(학생1.학번,수강4.학번)}$ 수강4

학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F

내부 조인

[요약] 관계 대수

❖ 외부 조인 (오른쪽 외부 조인)

-- 오른쪽 외부 조인(left outer join)

select *

from 학생1 RIGHT OUTER JOIN 수강4

on 학생1.학번 = 수강4.학번;

완전 외부 조인(\bowtie)

왼쪽 외부 조인(\bowtie)

오른쪽 외부 조인(\bowtie)

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

수강4

학번	과목번호	평가학점
s001	c002	A
s002	c001	F
s001	c004	B
s004	c005	C

학생1 \bowtie (학생1.학번,수강4.학번) 수강4



학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F
s004	NULL	NULL	NULL	c005	C

학생1 \bowtie N(학생1.학번,수강4.학번) 수강4

학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F

내부 조인

[요약] 관계 대수

❖ 외부 조인 (완전 외부 조인)

-- 완전 외부 조인(left outer join)

완전 외부 조인(\bowtie)

왼쪽 외부 조인(\ltimes)

오른쪽 외부 조인(\rtimes)

학생1

학번	이름	학년	성별
s001	김연아	4	여
s002	홍길동	1	남
s003	이승엽	3	남

수강4

학번	과목번호	평가학점
s001	c002	A
s002	c001	F
s001	c004	B
s004	c005	C

학생1 \bowtie (학생1.학번,수강4.학번) 수강4



학생1 \bowtie N(학생1.학번,수강4.학번) 수강4

학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F
s003	이승엽	3	남	NULL	NULL
s004	NULL	NULL	NULL	c005	C

학번	이름	학년	성별	과목번호	평가학점
s001	김연아	4	여	c002	A
s001	김연아	4	여	c004	B
s002	홍길동	1	남	c001	F

내부 조인

[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

[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	π 학년,학과(학생)	5	DIFFERENCE
6	CARTESIAN	7	σ 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

[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);
```


[Quiz]

12주차

담당교수: 김희숙
(jasmin11@hanmail.net)

[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)

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

- 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