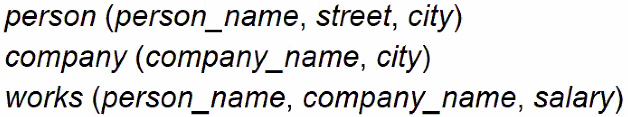
**Discussion 02 – cont.**

2-8

Relational Algebra는 procedural language이다. Procedural language는 데이터를 얻는 방법, 즉 순서를 정의하는데 relational algebra에서 데이터를 얻기 위해선 operation을 특정한 순서대로 정의해야 하기 때문에 procedural language의 정의와 일치한다.



2-9. Find all names of persons

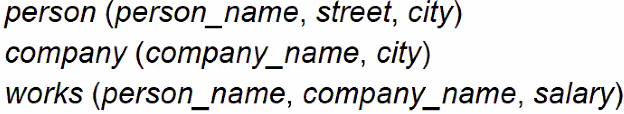
Π \_person\_name(person) U Π \_person\_name(works)

2-10. Find the names of persons who live in “Seoul”

Π \_person\_name(σ\_city=”Seoul”(person))

2-11. Find the names of persons who work for “SNU”

Π \_person\_name(σ\_company\_name=”SNU”(works))



2-12. Find all cities in the database

Π\_city(person) U Π\_city(company)

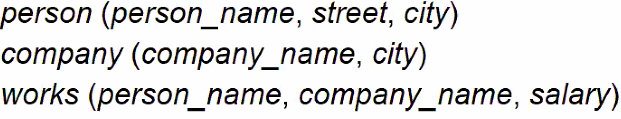
2-13. Find the names of people who do not work

Π\_person\_name(person) - Π\_person\_name(works)

2-14. Find names of people who work for “SNU” and earn more than 1,000,000

Π\_person\_name(σ\_company\_name=”SNU”\_and\_salary>=1000000(works))

**Discussion 06**



6-1. Find company names located in the city where “SNU” is located.

이름이 SNU인 회사의 city 이름을 S라 하면 S = Π\_city(σ\_company\_name=”SNU”(company))

그러면 Π\_company\_name(σ\_company\_name=S(company))

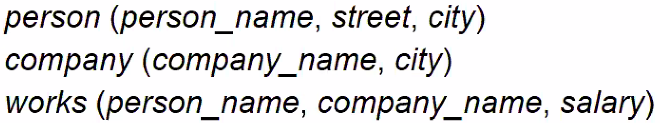
🡪 위처럼 하면 안되고 교수님 설명 들어보면 위의 S와 company relation을 cartesian product를 한 뒤 company.city = S.city 이런 식으로 select 해서, company\_name을 product 해야 한다고 한다. => Π \_company.company\_name(σ\_ company.city = S.city(company X S))

.6-2. Find names and addresses of employees who work for companies located in “Seoul”.

M = σ\_city=”Seoul”(company) X person X works

N = σ\_works.company\_name=company.company\_name\_AND\_works.person\_name=person.person\_name(M)

Π\_person.person\_name, person.street(N)



같은 relation을 두 번 참조할 경우 Rename이라는 operator 사용해서 이름을 바꿔줘야 함.

6-3. Find pairs of person names who live in the same city.

6-4. Find value of the largest salary.

연습해보기..