DATABASE SYSTEMS

Dr. Noha Nagy

Lecture 12

Relational Query Languages

- Languages for describing queries on a relational database
- *Structured Query Language* (SQL)
  - Predominant application-level query language
  - Declarative
- *Relational Algebra*
  - Intermediate language used within DBMS
  - Procedural

Relational Algebra Operations

- Unary Operations
  - Selection σ (sigma))
  - Projection π (pi))
  - Rename ρ (rho))
- Binary Operations
  - Union ∪
  - Intersection ∩
  - Set difference –
  - Cartesian product x

Select Operator

- Produce table containing subset of **rows** of argument table satisfying condition

$$\sigma_{condition}\, relation$$

- Example:

Person

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

$\sigma_{Hobby='stamps'}(Person)$

Id	*Name*	*Address*	*Hobby*
1123	John	23 Main	stamps
9876	Bart	Pine St	stamps

Selection Condition - Examples

- $\sigma_{Id>3000 \text{ Or } Hobby='hiking'}$ (Person)
- $\sigma_{Id>3000 \text{ AND } Id<3999}$ (Person)
- $\sigma_{\text{NOT}(Hobby='hiking')}$ (Person)
- $\sigma_{Hobby\neq'hiking'}$ (Person)

STUDENT

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
456	Maha	Nasr City	CE	1.9
789	Ahmad	Haram	Arch	2.7
341	Noha	Dokki	EE	1.0

$$\sigma_{\text{Address = “Dokki” and Major= “EE”}}(\text{STUDENT})$$

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
341	Noha	Dokki	EE	1.0

STUDENT

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
456	Maha	Nasr City	CE	1.9
789	Ahmad	Dokki	Arch	2.7
341	Noha	Dokki	EE	1.0

$$\sigma_{\text{Address = “Dokki” or Major= “EE”}}(\text{STUDENT})$$

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
789	Ahmad	Dokki	Arch	2.7
341	Noha	Dokki	EE	1.0

Unary Relational Operations: SELECT (contd.)

- SELECT Operation Properties
  - SELECT σ is commutative:
    - $\sigma_{<condition1>}(\sigma_{<condition2>}(R)) = \sigma_{<condition2>}(\sigma_{<condition1>}(R))$
  - Because of commutativity property, a cascade (sequence) of SELECT operations may be applied in **any order:**
    - $\sigma_{<cond1>}(\sigma_{<cond2>}(\sigma_{<cond3>}(R)) = \sigma_{<cond2>}(\sigma_{<cond3>}(\sigma_{<cond1>}(R))))$
  - A cascade of SELECT operations may be replaced by a single selection with a conjunction of all the conditions:

    $\sigma_{<cond1>}(\sigma_{<cond2>}(\sigma_{<cond3>}(R)) = \sigma_{<cond1> \text{ AND } <cond2> \text{ AND } <cond3>}(R)))$

Unary Relational Operations: PROJECT

- PROJECT Operation is denoted by π (pi)
- This operation keeps certain c*olumns* (attributes) from a relation and discards the other columns.
- Example: To list each employee's first and last name and salary, the following is used:

  $\pi_{LNAME,\ FNAME,SALARY}(EMPLOYEE)$

Project Operator

- Produces table containing subset of columns of argument table

$$\Pi_{attribute\ list}(relation)$$

- Example:

Person

Id	*Name*	*Address*	*Hobby*
1123	John	123 Main	stamps
1123	John	123 Main	coins
5556	Mary	7 Lake Dr	hiking
9876	Bart	5 Pine St	stamps

$\Pi_{Name,Hobby}$(Person)

Name	*Hobby*
John	stamps
John	coins
Mary	hiking
Bart	stamps

PROJECT Operation π

STUDENT

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
456	Maha	Nasr City	CE	1.9
789	Ahmad	Haram	Arch	2.7
341	Noha	Dokki	EE	1.0

$\pi_{\text{ST-ID, Major}}$ (STUDENT)

ST-ID	Major
123	EE
456	CE
789	Arch
341	EE

π_{Major} (STUDENT)

Major
EE
CE
Arch

RA returns distinct tuples

Unary Relational Operations: PROJECT (contd.)

- PROJECT Operation Properties
  - The number of tuples in the result of projection $\pi_{<list>}(R)$ is always less or equal to the number of tuples in R (WHEN EQUAL???)
    - If the list of attributes includes a *key* of R, then the number of tuples in the result of PROJECT is *equal* to the number of tuples in R
  - PROJECT is ***NOT*** commutative
    - $\pi_{<list1>}(\pi_{<list2>}(R)) = \pi_{<list1>}(R)$ as long as <list2> contains the attributes in <list1>

- Find names and GPAs of either EE or CE students where the GPA is less than 2.0.

STUDENT

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	1.2
456	Maha	Nasr City	CE	1.9
789	Ahmad	Haram	Arch	2.7
341	Noha	Dokki	EE	1.0

$$\pi_{\text{Name, GPA}} (\sigma_{\text{(Major = "EE" or Major = "CE") and GPA < 2.0}} (\text{STUDENT}))$$

Name	GPA
Ali	1.2
Maha	1.9
Noha	1.0

Single expression versus sequence of relational operations (Example)

- Retrieve the first name, last name, and salary of all employees who work in department number 5.
  - $\pi_{FNAME, LNAME, SALARY}(\sigma_{DNO=5}(EMPLOYEE))$
- OR We can explicitly show the *sequence of operations*, giving a name to each intermediate relation:
  - DEP5_EMPS $\leftarrow \sigma_{DNO=5}(EMPLOYEE)$
  - RESULT $\leftarrow \pi_{FNAME, LNAME, SALARY}$ (DEP5_EMPS)

STUDENT

ST-ID	Name	Address	Major	GPA
123	Ali	Dokki	EE	3.2
456	Maha	Nasr City	CE	1.9
789	Reda	Dokki	Arch	2.7
341	Noha	Dokki	EE	1.0

PROF

P-ID	PName	ADD	WorksIn	Rank
P71	Ali	Shobra	Comm	Full
P08	Karem	Dokki	NULL	Full
P11	Magda	Agoza	CE	Assoc
P91	Nagy	Dokki	EE	Full

STUDENT X PROF

ST-ID	Name	Address	Major	GPA	P-ID	PName	ADD	WorksIn	Rank
123	Ali	Dokki	EE	3.2	P71	Ali	Shobra	Comm	Full
123	Ali	Dokki	EE	3.2	P08	Karem	Dokki	NULL	Full
123	Ali	Dokki	EE	3.2	P11	Magda	Agoza	CE	Assoc
123	Ali	Dokki	EE	3.2	P91	Nagy	Dokki	EE	Full
456	Maha	Nasr City	CE	1.9	P71	Ali	Shobra	Comm	Full
456	Maha	Nasr City	CE	1.9	P08	Karem	Dokki	NULL	Full
456	Maha	Nasr City	CE	1.9	P11	Magda	Agoza	CE	Assoc
456	Maha	Nasr City	CE	1.9	P91	Nagy	Dokki	EE	Full
789	Reda	Dokki	Arch	2.7	P71	Ali	Shobra	Comm	Full
789	Reda	Dokki	Arch	2.7	P08	Karem	Dokki	NULL	Full
789	Reda	Dokki	Arch	2.7	P11	Magda	Agoza	CE	Assoc
789	Reda	Dokki	Arch	2.7	P91	Nagy	Dokki	EE	Full
341	Noha	Dokki	EE	1.0	P71	Ali	Shobra	Comm	Full
341	Noha	Dokki	EE	1.0	P08	Karem	Dokki	NULL	Full
341	Noha	Dokki	EE	1.0	P11	Magda	Agoza	CE	Assoc
341	Noha	Dokki	EE	1.0	P91	Nagy	Dokki	EE	Full

Relational Algebra Operations from Set Theory: CARTESIAN PRODUCT (cont.)

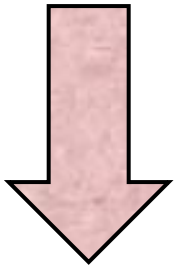
- **Generally, CARTESIAN PRODUCT is not a meaningful operation**
  - Can become meaningful when followed by other operations

$\sigma_{\text{Address = ADD}}$ (STUDENT X PROF) □

ST-ID	Name	Address	Major	GPA	P-ID	PName	ADD	WorksIn	Rank
123	Ali	Dokki	EE	3.2	P71	Ali	Shobra	Comm	Full
123	Ali	Dokki	EE	3.2	P08	Karem	Dokki	NULL	Full
123	Ali	Dokki	EE	3.2	P11	Magda	Agoza	CE	Assoc
123	Ali	Dokki	EE	3.2	P91	Nagy	Dokki	EE	Full
456	Maha	Nasr City	CE	1.9	P71	Ali	Shobra	Comm	Full
456	Maha	Nasr City	CE	1.9	P08	Karem	Dokki	NULL	Full
456	Maha	Nasr City	CE	1.9	P11	Magda	Agoza	CE	Assoc
456	Maha	Nasr City	CE	1.9	P91	Nagy	Dokki	EE	Full
789	Reda	Dokki	Arch	2.7	P71	Ali	Shobra	Comm	Full
789	Reda	Dokki	Arch	2.7	P08	Karem	Dokki	NULL	Full
789	Reda	Dokki	Arch	2.7	P11	Magda	Agoza	CE	Assoc
789	Reda	Dokki	Arch	2.7	P91	Nagy	Dokki	EE	Full
341	Noha	Dokki	EE	1.0	P71	Ali	Shobra	Comm	Full
341	Noha	Dokki	EE	1.0	P08	Karem	Dokki	NULL	Full
341	Noha	Dokki	EE	1.0	P11	Magda	Agoza	CE	Assoc
341	Noha	Dokki	EE	1.0	P91	Nagy	Dokki	EE	Full

JOIN Operation

$\sigma_{X.a = Y.a}(X \times Y)$ □



$Y \bowtie_{X.a = Y.a} X$

⋈ JOIN Operation

X.a = Y.a is called the JOIN Condition

Examples of Queries in Relational Algebra

- **Q1: Retrieve the name and address of all employees who work for the 'Research' department.**
- **Employee(Eid,Fname,Lname,address,salary,Dno)**
- **Department(Dnumber,Dname)**

$RESEARCH\_DEPT \leftarrow \sigma_{DNAME='Research'}(DEPARTMENT)$

$RESEARCH\_EMPS \leftarrow (RESEARCH\_DEPT \bowtie_{DNUMBER=DNO} EMPLOYEE)$

$RESULT \leftarrow \pi_{FNAME, LNAME, ADDRESS}(RESEARCH\_EMPS)$

How to transform from SQL to RA

Select name, age

From student

Where depno=1

π

σ

$\pi_{\text{name, age}}\ \sigma_{\text{depno=1}}\ \textbf{(student)}$

How to transform from SQL to RA

Student(sid,name,age,dno)

Department (Did, depname)

Select depname

From student, department

Where dno=did

And age>30

π

σ

$$\pi_{name}\,\sigma_{age>30}\,(student \underset{Dno=did}{\bowtie} department)$$

$$\pi_{name}\,\pi_{salary}\,\pi_{age}\,\mathbf{student}$$

Set operation

Relations *r*, *s*:

A	B
α	1
α	2
β	1

r

A	B
α	2
β	3

s

r ∪ s:

A	B
α	1
α	2
β	1
β	3

r − *s*:

A	B
α	1
β	1

Outer Join

instructor ⋈ *teaches*

ID	*name*	*dept_name*	*course_id*
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201

- Left Outer Join

  instructor ⟕ *teaches*

ID	*name*	*dept_name*	*course_id*
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	*null*

Outer Join

- Right Outer Join

  instructor ⟖ *teaches*

ID	*name*	*dept_name*	*course_id*
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
76766	null	null	BIO-101

- Full Outer Join

  instructor ⟗ *teaches*

ID	*name*	*dept_name*	*course_id*
10101	Srinivasan	Comp. Sci.	CS-101
12121	Wu	Finance	FIN-201
15151	Mozart	Music	*null*
76766	null	null	BIO-101

Aggregate function

"Find the total amount owed to the credit company."

$\mathcal{G}_{\textbf{sum}(balance)}(credit\_acct)$

4275

cred_id	limit	balance
C-273	2500	150
C-291	750	600
C-304	15000	3500
C-313	300	25

credit_acct

"Find the maximum available credit of any account."

$\mathcal{G}_{\textbf{max}(available\_credit)}(\Pi_{(limit - balance)\ \textbf{as}\ available\_credit}(credit\_acct))$

11500

Divide Operation

- Find suppliers that supply all parts

```
SELECT * FROM suppliers as s
WHERE NOT EXISTS (( SELECT p.pid FROM parts as p )
EXCEPT
 (SELECT sp.pid FROM supplies sp WHERE sp.sid = s.sid ) );
```

sid (integer)	pid (integer)
101	1
102	1
101	3
103	2
102	2
102	3
102	4
102	5

pid (integer)
1
2
3
4
5