Relational Algebra (Part I)

R & G, Chapter 4

Databases: the continuing saga

- What we have learned:
  - Conceptually model data needs using ER diagrams
  - Translate ER diagrams to relational schemas
  - Create schemas and insert records into relational tables

We're almost ready to use SQL to query it, but first…

Relational Query Languages

- *Query languages:* Allow manipulation and retrieval of data from a database.
- Relational model supports simple, powerful QLs:
  - Strong formal foundation based on logic.
  - Allows for optimization.
- Query Languages != programming languages!
  - QLs not intended to be used for complex calculations.
  - QLs support easy, efficient access to large data sets.

Formal Relational Query Languages

Two mathematical Query Languages form the basis for "real" languages (e.g. SQL), and for implementation:

Relational Algebra: More operational, very useful for representing execution plans.

√

Relational Calculus: Let users describe what they want, rather than how to compute it. (Non-procedural, *declarative*.)

We only cover Relational Algebra in CS442

Relational Algebra: 5 Basic Operations

Unary operators (only one table as input)

1. *Selection* (σ)
2. *Projection* (π)

Set operations (two compatible tables as input)

3. *Set-difference* (—)
4. *Union* (U)

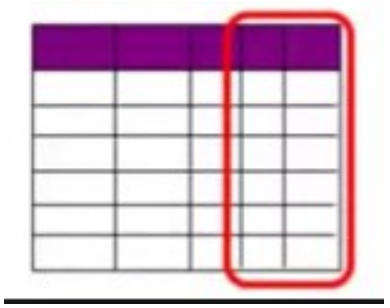
Non-set operation (two tables as input)

5. Cross-product operation (X)

Since each operation returns a relation, operations can be *composed!*

Projection (π)

- Projection (π): pick certain *columns* (i.e., attributes) for output
- Notation: $\pi_{A1, A2 \ldots Ak}(r)$
  - r: a relational table;
  - $A_1, \ldots, A_k$: attributes in the output
- Output: return attributes $A_1, \ldots, A_k$ *in r*
- Outputs may contain *duplicate records.* (How do they arise?)



Projection Examples

Sid	Sname	Rating	Age
28	Yuppy	9	35
31	Lubber	8	55
44	Guppy	5	35
58	Rusty	10	35

Sailor

$\pi_{sname,rating}(Sailor)$

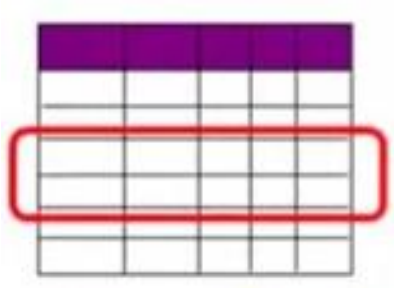
Sname	Rating
Yuppy	9
Lubber	8
Guppy	5
Rusty	10

$\pi_{age}(Sailor)$

Age
35
55
35
35

Selection (σ)

- Selection (**σ**): pick certain rows that satisfy specific conditions for output



- Notation: $\sigma_p(r)$
  - r: a relational table
  - p: *selection condition*
    - **predicates:**

      <attribute> ***op*** <attribute> (e.g., salary < bonus)

      OR

      <attribute> ***op*** <constant> (e.g., name = 'smith')

      *(**op**: <, <=, =, ≠, >=, >)*
    - **Connectives of predicates**: ∧ (and) and ∨ (or)
      - *E.g.: name = 'Smith'* ∧age =20

Selection (σ)

- Output of *selection* operator $\sigma_{p}(r)$
  - Schema: the same as the schema of the input relation r
  - Instance:
    - It outputs the records that satisfy the selection condition.
    - **Question**: does the selection results contain duplicates?

Selection (σ) Example

Sid	Sname	Rating	Age
28	Yuppy	9	35
~~31~~	~~Lubber~~	~~8~~	~~55~~
~~44~~	~~Guppy~~	~~5~~	~~35~~
58	Rusty	10	35

$$\sigma_{rating>8}(S2)$$

Order between Selection (σ) and Projection (π)

- Projection π is always placed before selection σ.
- Format (put σ closest to r, and π farthest from r)

$$\pi_{A1,\, A2 \ldots Ak}\, \sigma_{p}(r)$$

- Evaluation order: selection σ first, projection π last.

Example of Selection (σ) and Projection (π) Together

Sid	Sname	Rating	Age
28	Yuppy	9	35
31	Lubber	8	55
44	Guppy	5	35
58	Rusty	10	35

$\sigma_{rating>8}(S2)$

Sname	Rating
Yuppy	9
Rusty	10

$\pi_{sname,Rating}\ \sigma_{rating>8}(S2)$

Example (I)

Bid	Bname	Color
101	Interlake	Blue
102	Interlate	Red
103	Clipper	Green
104	Marine	red

Boats

- Write the relational algebra expression to find the color of the boat named "Marine"

Example (II)

Sid	Sname	Rating	Age
22	Dustin	7	45
31	Lubber	8	55
58	Rusty	10	35

Sailor

- Write the relational algebra expression to find the name of the sailors whose rating is at least 8.

- Write the relational algebra expression to find the name and rating of the sailors whose rating is at least 8 and whose age is at most 40.

Union (U) and Set-Difference (-)

- Set operations take two input relations R1 and R2
- R1 and R2 must be *union-compatible*:
  - Union-compatible = same schema
    - Same number of attributes;
    - Corresponding attributes have the same data type.

Union (U)

- **Notation:** $R \cup S$
- Returns a relation instance containing all tuples that in either R or S (or both)
- Symmetry of U: R U S = S U R

Sid	Sname	Rating	Age
22	Dustin	7	45
31	Lubber	8	55
58	Rusty	10	35

S1

Sid	Sname	Rating	Age
28	Yuppy	9	35
31	Lubber	8	55
44	Guppy	5	35
58	Rusty	10	35

S2

Sid	Sname	Rating	Age
22	Dustin	7	45
31	Lubber	8	55
58	Rusty	10	35
28	Yuppy	9	35
44	Guppy	5	35

S1 U S2

Set Difference (-)

- Notation: R - S
- returns a relation instance containing all tuples that in R but not S.

Sid	Sname	Rating	Age
22	Dustin	7	45.0
31	Lubber	8	55.5
58	Rusty	10	35.5

S1

Sid	Sname	Rating	Age
28	Yuppy	9	35.0
31	Lubber	8	55.5
44	Guppy	5	35.0
58	Rusty	10	35.5

S2

Sid	Sname	Rating	Age
22	Dustin	7	45.0

S1 – S2

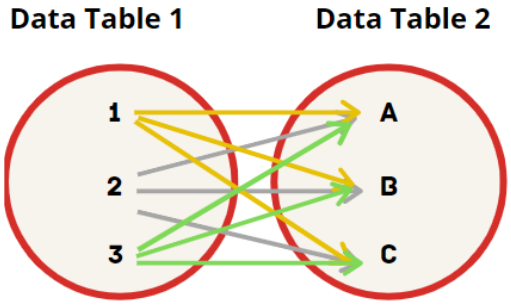
Sid	Sname	Rating	Age
28	Yuppy	9	35.0
44	Guppy	5	35.0

S2 – S1

Q1: Is there any duplicate in the set difference result?
Q2: Is the set difference (-) operator symmetric (i.e., R- S = S- R)?

Cross-Product (X) Operator

- Notation: **R X S**
- Output:
  - Schema: the attributes of R followed by the attributes of S, in order.
    - If R and S contain the same attribute A, the output schema includes both R.A and S.A
  - Instances in the output:
    - A Cartesian product of R and S
    - Pair each tuple of R with each tuple of S.



- S and R DO NOT have to be compatible
- **Question:** Will there be any duplicate in the cross-product result?

Cross Product Example

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

R

Sid	Sname	Rating	Age
22	Dustin	7	45
31	Lubber	8	55
58	Rusty	10	35

S

R X S =

R.Sid	Bid	Day	S.Sid	Sname	Rating	Age
22	101	10/10/96	22	Dustin	7	45
22	101	10/10/96	31	Lubber	8	55
22	101	10/10/96	58	Rusty	10	35
58	103	11/12/96	22	Dustin	7	45
58	103	11/12/96	31	Lubber	8	55
58	103	11/12/96	58	Rusty	10	35

Cross-Product

- Assume S contains n1 rows, and k1 attributes, and R contains n2 rows, and k2 attributes

- Questions:
  - How many attributes are in S X R?
  - How many tuples are in S X R?

Summary

Operator	# of input instances	Duplicates in output	Union-compatible	Symmetric
Projection	1	Y	N/A	N/A
Selection	1	N	N/A	N/A
Union	2	N	Y	Y
Set difference	2	N	Y	N
Cross-product	2	N	N	Y