

Lab 3

1. RELATIONAL ALGEBRA

Figure 1 shows some of the commonly used relational algebra operators.

Symbol (Name)	Example of Use
σ (Selection)	$\sigma \text{ salary} \geq 85000 \text{ (instructor)}$ Return rows of the input relation that satisfy the predicate.
Π (Projection)	$\Pi \text{ ID, salary (instructor)}$ Output specified attributes from all rows of the input relation. Remove duplicate tuples from the output.
\times (Cartesian Product)	$\text{instructor} \times \text{department}$ Output all pairs of rows from the two input relations (regardless of whether or not they have the same values on common attributes)
\cup (Union)	$\Pi \text{ name (instructor)} \cup \Pi \text{ name (student)}$ Output the union of tuples from the <i>two</i> input relations.
$-$ (Set Difference)	$\Pi \text{ name (instructor)} - \Pi \text{ name (student)}$ Output the set difference of tuples from the two input relations.
\bowtie (Natural Join)	$\text{instructor} \bowtie \text{department}$ Output pairs of rows from the two input relations that have the same value on all attributes that have the same name.

FIGURE 1. Relational algebra operators.

Q1. Having the relation schema below for committee member management, please answer the questions below using relational algebra.

professor (prof_name, dept_name)
department (dept_name, building)
committee (comm_name, prof_name)

- Project committee names having members from the "ECE" department.
- Project the name of professors sharing at least one committee with Prof. "Smith".

Q2. Having the relation schema below for medical records, please answer the questions below using relational algebra.

patient(p_id, p_name, address)
 doctor(d_id, d_name, hospital)
 medicine(m_id, m_name)
 appointment(a_id, p_id, d_id, date)
 prescription(a_id, m_id)

- Show those medicines that have been listed at least once in the prescriptions of Dr. "Luis".
- Project the name and address of patients who have visited any doctors of "Garcia" hospital at least once.
- Project the name of medicines that have been used in at least one prescription.
- Assuming that all the names are unique, list the name of doctors who have visited another doctor. (In a scenario that doctor a as a patient goes to doctor b 's office, we want doctor b 's name.)

2. IMPLEMENTATION

Q3. In this example, we deal with a Boat Reservation System. Please follow the step-by-step instructions using the code provided in instructions.sql.

- Make sure you are on the right database using `current_database()` function.
- Create sufficient tables for the schema below:

sailor(s_id, s_name, age, rating)
 boat(b_id, b_name, color)
 reserve(s_id, b_id, r_date)

For convenience, the code to create a table for Sailors is provided. Please create the tables one by one.

- Insert sailors and boats instances as Tables 1 and 2 below. The command to insert the first row is provided.

s_id	s_name	age	rating
1	James	45	10
2	Mary	36	5
3	Robert	50	9
4	Patricia	34	8
5	John	24	2
6	Jennifer	20	2
7	Michael	20	10
8	Linda	35	3
9	David	58	10
10	Elizabeth	55	5

TABLE 1. Sailors

b_id	b_name	color
1	Serendipity	Blue
2	Imagination	Red
3	Liberty	Blue
4	Wanderlust	Green
5	Gale	Blue
6	Zephyr	Green

TABLE 2. Boats

- Make sure that the rows are inserted correctly by selecting all the values from each table.

- e) Insert reservation records based on the code provided.
- f) List the IDs of sailors who had at least one trip on the boat with ID 3. You only need *reserve* table for this query.
- g) Now list the name of the sailors that you listed in previous query. Their names are listed in *sailor* table. Use your previous query to check the WHERE condition.
- h) We have two new Sailors who have not been on a trip yet. List their names.
- i) List all the sailor IDs who have been on a trip with a red or green boat. Use keyword DISTINCT to remove duplicates from your result.
- j) We have a sailor who likes red boats and does not like green boats. Write a query to find their name. You may find the keyword EXCEPT handy to implement set difference.