Relational Algebra

Relational algebra is a **procedural query language** which works on relational models. Procedural query language tells what data to be retrieved and how to be retrieved.

A fundamental property is that every operator in the algebra accepts (one or two) relation instances as arguments and returns a relation instance as the result.

A relational algebra expression is recursively defined to be a relation, a unary algebra operator applied to a single expression, or a binary algebra operator applied to two expressions.

The fundamental operations in the relational algebra are **select, project, union, set difference, Cartesian product, and rename.**
The select, project, and rename operations are called unary operations, because they operate on one relation.
The other three operations operate on pairs of relations and are, therefore, called binary operations.

Set Operation

1. **Union**
2. **Intersection**
3. **Set difference**
4. **Cartesian product**

Union

- Union of two relations R and S (R U S) defines a relation that contains all the tuples of R, or S, or both R and S, duplicate tuples being eliminated.
- R and S must be union-compatible.
- If R and S have I and J tuples, respectively, union is obtained by concatenating them into one relation with a maximum of (I + J) tuples.
- UNION is symbolized by ∪ symbol.

Example:

Graduates

Number	Surname	Age
7274	Robinson	37
7432	O'Malley	39
9824	Darkes	38

Managers

Number	Surname	Age
9297	O'Malley	56
7432	O'Malley	39
9824	Darkes	38

Graduates ∪ Managers

Number	Surname	Age
7274	Robinson	37
7432	O'Malley	39
9824	Darkes	38
9297	O'Malley	56

Intersection

- The intersection of two relations R and S(R ∩ S), defines a relation consisting of the set of all tuples that are in both R and S.
- R and S must be union-compatible.
- Represented using basic operations: R ∩ S = R – (R – S)

Example:

Graduates

Number	Surname	Age
7274	Robinson	37
7432	O'Malley	39
9824	Darkes	38

Managers

Number	Surname	Age
9297	O'Malley	56
7432	O'Malley	39
9824	Darkes	38

Graduates ∩ Managers

Number	Surname	Age
7432	O'Malley	39
9824	Darkes	38

Set Difference

- The difference of two relations R and S (R – S), define a relation consisting of the tuples that are in relation R, but not in S.
- R and S must be union-compatible.
- It is denoted by(-).
- Represented using basic operations: R -S

Example:

Graduates

Number	Surname	Age
7274	Robinson	37
7432	O'Malley	39
9824	Darkes	38

Managers

Number	Surname	Age
9297	O'Malley	56
7432	O'Malley	39
9824	Darkes	38

Graduates - Managers

Number	Surname	Age
7274	Robinson	37

Cartesian product

- The Cartesian product of two relations R and S (R X S) , defines a relation between every tuple of relation R with every tuple of relation S.
- It is denoted by (X).
- Represented using basic operations: R X S

Example:

Student

Sr No	Name	Grade
1	Andrew	D
2	Robin	B

Data

Seat No	Age
18	25
11	21

Student x Data

Sr No	Name	Grade	Seat No	Age
1	Andrew	D	18	25
1	Andrew	D	11	21
2	Robin	B	18	25
2	Robin	B	11	21

Unary Relational operations

1. Select
2. Project
3. Rename

Select Operation

- The select operation is performed to select certain rows or tuples of a table, so it performs its action on the table horizontally.,
- The tuples are selected through this operation using a predicate or condition.
- It works on a single table and takes rows that meet a specified condition, copying them into a new table.
- Denoted by lower Greek letter sigma (σ).

Relation: σ predicate(R)

Example of Selection:

Employees

Surname	FirstName	Age	Salary
Smith	Mary	25	2000
Black	Lucy	40	3000
Verdi	Nico	36	4500
Smith	Mark	40	3900

$\sigma_{\text{Age<30 v Salary>4000}}$ **(Employees)**

Surname	FirstName	Age	Salary
Smith	Mary	25	2000
Verdi	Nico	36	4500

In selection operation the comparison operators like <, >, =, <=, >=, <> can be used in the predicate

Project Operation

- The Select operation operates horizontally on the table. Conversely, the Project operator works on a single table vertically.
- It is a unary operation that returns a relation that includes a **subset of the attributes** of the operand.
- Since the relation is a set, any duplicate rows are eliminated.
- Projection is represented by a Greek letter (Π) .

Example of Projection:

Employees

Surname	FirstName	Department	Head
Smith	Mary	Sales	De Rossi
Black	Lucy	Sales	De Rossi
Verdi	Mary	Personnel	Fox
Smith	Mark	Personnel	Fox

$\pi_{\text{Surname, FirstName}}(\text{Employees})$

Surname	FirstName
Smith	Mary
Black	Lucy
Verdi	Mary
Smith	Mark

Rename

- This is a unary operator which changes attribute names for a relation without changing any values.
- Renaming removes the limitations associated with set operators.
- Representation: ρ old name(R) → New Name(R)

Example: ρ Father->Parent(Paternity)

Paternity

Father	Child
Adam	Cain
Adam	Abel
Abraham	Isaac
Abraham	Ishmael

$\rho_{Father \rightarrow Parent}$(Paternity)

Parent	Child
Adam	Cain
Adam	Abel
Abraham	Isaac
Abraham	Ishmael

Join Operation

- The JOIN operation, denoted by ⋈, is used to combine related tuples from two relations into single "longer" tuples.
- The join operator allows the combination of two relations to form a single new relation.
- The JOIN operation can be specified as a CARTESIAN PRODUCT operation followed by a SELECT operation.

Natural Join

An Equijoin of the two relations R and S over all common attributes x. One occurrence of each common attribute is eliminated from the result.
Hence the degree is the sum of the degrees of the relations R and S less the number of attributes in x

r_1

Employee	Department
Smith	sales
Black	production
White	production

r_2

Department	Head
production	Mori
sales	Brown

$r1 \bowtie r2$

Employee	Department	Head
Smith	sales	Brown
Black	production	Mori
White	production	Mori

Theta Join

Defines a relation that contains tuples satisfying the predicate F from the Cartesian product of R and S.

The predicate F is of the form R.ai θ S.bi where θ may be one of the comparison operators (<, ≤, >, ≥, =, ≠).