

Introduction Material:

What is Relational Calculus?

Relational calculus is a query language which is non-procedural, and instead of algebra, it uses mathematical predicate calculus. The relational calculus is not the same as that of differential and integral calculus in mathematics but takes its name from a branch of symbolic logic termed as predicate calculus. When applied to databases, it is found in two forms. These are

- Tuple relational calculus which was originally proposed by Codd in the year 1972 and
- Domain relational calculus which was proposed by Lacroix and Pirotte in the year 1977

In first-order logic or predicate calculus, a predicate is a truth-valued function with arguments. When we replace with values for the arguments, the function yields an expression, called a proposition, which will be either true or false.

Example:

For example, steps involved in listing all the employees who attend 'Networking' Course would be:

```
SELECT the tuples from COURSE relation with COURSENAME =  
'NETWORKING'
```

PROJECT the COURSE_ID from above result

SELECT the tuples from EMP relation with COURSE_ID resulted above.

Tuple Relational Calculus

In the tuple relational calculus, you will have to find tuples for which a predicate is true. The calculus is dependent on the use of tuple variables. A tuple variable is a variable that 'ranges over' a named relation: i.e., a variable whose only permitted values are tuples of the relation.

Example:

For example, to specify the range of a tuple variable S as the Staff relation, we write:

```
Staff(S)
```

To express the query 'Find the set of all tuples S such that F(S) is true,' we can write:

$$\{S \mid F(S)\}$$

Here, F is called a formula (well-formed formula, or wff in mathematical logic). For example, to express the query 'Find the staffNo, fName, lName, position, sex, DOB, salary, and branchNo of all staff earning more than £10,000', we can write:

$$\{S \mid \text{Staff}(S) \wedge S.\text{salary} > 10000\}$$

Example:

$$\{t \mid \text{TEACHER}(t) \text{ and } t.\text{SALARY} > 20000\}$$

- It implies that it selects the tuples from the TEACHER in such a way that the resulting teacher tuples will have the salary greater than 20000. This is an example of selecting a range of values.

$$\{t \mid \text{TEACHER}(t) \text{ AND } t.\text{DEPT_ID} = 6\}$$

- T select all the tuples of teachers name who work under Department 8. Any tuple variable with 'For All' (?) or 'there exists' (?) condition is termed as a bound variable. In the last example, for any range of values of SALARY greater than 20000, the meaning of the condition does not alter. Bound variables are those ranges of tuple variables whose meaning will not alter if another tuple variable replaces the tuple variable.

In the second example, you have used DEPT_ID= 8 which means only for DEPT_ID = 8 display the teacher details. Such variable is called free variable. Any tuple variable without any 'For All' or 'there exists' condition is called Free Variable.

Domain Relational Calculus

In the tuple relational calculus, you have use variables that have series of tuples in a relation. In the domain relational calculus, you will also use variables, but in this case, the variables take their values from domains of attributes rather than tuples of relations. A domain relational calculus expression has the following general format:

$$\{d1, d2, \dots, dn \mid F(d1, d2, \dots, dm)\} \quad m \geq n$$

where $d_1, d_2, \dots, d_n, \dots, d_m$ stand for domain variables and $F(d_1, d_2, \dots, d_m)$ stands for a formula composed of atoms.

Example:

select TCHR_ID and TCHR_NAME of teachers who work for department 8,
(where suppose - dept. 8 is Computer Application Department)

```
{<tchr_id, tchr_name=""> | <tchr_id, tchr_name=""> ? TEACHER ∧ DEPT_ID = 10}
```

Get the name of the department name where Karlos works:

```
{DEPT_NAME | <DEPT_NAME> ? DEPT ∧ ? DEPT_ID ( ? TEACHER ∧ TCHR_NAME = Karlos)}
```

It is to be noted that these queries are safe. The use domain relational calculus is restricted to safe expressions; moreover, it is equivalent to the tuple relational calculus which in turn is similar to the relational algebra.

Credits:

<https://www.w3schools.in/dbms/relational-calculus/>