Relational Calculus:

There is an alternate way of formulating queries known as Relational Calculus.

Relational calculus is a non-procedural query language.

In the non-procedural query language, the user is concerned with the details of how to obtain the end results.

The relational calculus tells what to do but never explains how to do.

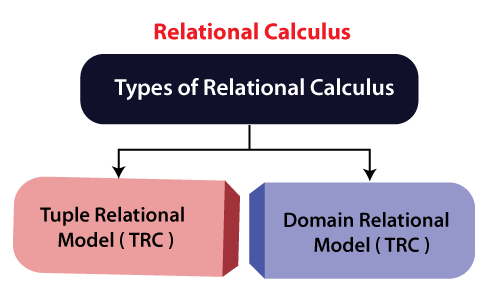
It is based on Predicate calculus, a name derived from branch of symbolic language.

A predicate is a truth-valued function with arguments.

On substituting values for the arguments, the function result in an expression called a proposition. It can be either true or false.

It is a tailored version of a subset of the Predicate Calculus to communicate with the relational database.

Most commercial relational languages are based on aspects of relational calculus including SQL-QBE and QUEL.(query language)



Tuple Relational Calculus (TRC)

It is a non-procedural query language which is based on finding a number of tuple variables also known as range variable for which predicate holds true. It describes the desired information without giving a specific procedure for obtaining that information. The tuple relational calculus is specified to select the tuples in a relation. In TRC, filtering variable uses the tuples of a relation. The result of the relation can have one or more tuples.

**Notation:**

{T | P (T)}   or {T | Condition (T)}

Where

**T** is the resulting tuples

**P(T)** is the condition used to fetch T.

**Output:** This query selects the tuples from the AUTHOR relation. It returns a tuple with 'name' from Author who has written an article on 'database'.

TRC (tuple relation calculus) can be quantified. In TRC, we can use Existential (∃) and Universal Quantifiers (∀).

1. { R| ∃T ∈ Authors(T.article='database' AND R.name=T.name)}

**Output:** This query will yield the same result as the previous one.

The second form of relation is known as Domain relational calculus. In domain relational calculus, filtering variable uses the domain of attributes. Domain relational calculus uses the same operators as tuple calculus. It uses logical connectives ∧ (and), ∨ (or) and ┓ (not). It uses Existential (∃) and Universal Quantifiers (∀) to bind the variable. The QBE or Query by example is a query language related to domain relational calculus.

**Notation:**

1. { a1, a2, a3, ..., an | P (a1, a2, a3, ... ,an)}

Where

**a1, a2** are attributes  
**P** stands for formula built by inner attributes

**For example:**

1. {< article, page, subject > |  ∈ javatpoint ∧ subject = 'database'}

**Output:** This query will yield the article, page, and subject from the relational javatpoint, where the subject is a database.

To retrieve the names of all employees who earn more than $50,000 per year, we can use the following TRC query:

{ t | Employees(t) ∧ t.Salary > 50000 }

Find the loan number, branch, and amount of loans greater than or equal to 10000 amount.

{t| t ∈ loan ∧ t[amount]>=10000}

Find the loan number for each loan of an amount greater or equal to 10000.

{t| ∃ s ∈ loan(t[loan number] = s[loan number]

∧ s[amount]>=10000)}

Differences between Relational Algebra and Relational Calculus:

Relational Algebra is a procedural language. In Relational Algebra, The order is specified in which the operations have to be performed. In [Relational Algebra](https://www.geeksforgeeks.org/introduction-of-relational-algebra-in-dbms/), frameworks are created to implement the queries. The basic operation included in relational algebra are:

**1.** Select (σ)

**2.** Project (Π)

**3.** Union (U)

**4.** Set Difference (-)

**5.** Cartesian product (X)

**6.** Rename (ρ)

**Relational Calculus:**

Relational Calculus is the formal query language. It is also known as **Declarative language**. In Relational Calculus, the order is not specified in which the operation has to be performed. Relational Calculus means what result we have to obtain.   
Relational Calculus has two variations:

1. [Tuple Relational Calculus (TRC)](https://www.geeksforgeeks.org/dbms-tupple-relational-calculus/)
2. [Domain Relational Calculus (DRC)](https://www.geeksforgeeks.org/dbms-domain-relational-calculus/)

|  | **Basis of Comparison** |  |
| --- | --- | --- |
| **1.** | **Language Type** | **Relational Algebra :**It is a Procedural language. | **Relational Calculus** is a Declarative (non-procedural) language. |
| **2.** | **Procedure** | Relational Algebra means how to obtain the result. | Relational Calculus means what result we have to obtain. |
| **3.** | **Order** | In Relational Algebra, the order is specified in which the operations have to be performed. | In Relational Calculus, the order is not specified. |
| **4.** | **Domain** | Relational Algebra is independent of the domain. | Relation Calculus can be domain-dependent because of domain relational calculus. |
| **5.** | **Programming language** | Relational Algebra is nearer to a programming language. | Relational Calculus is not nearer to programming language but to natural language. |
| **6.** | **Inclusion in SQL** | The SQL includes only some features from the relational algebra. | SQL is based to a greater extent on the tuple relational calculus. |
| **7.** | **Relationally completeness** | Relational Algebra is one of the languages in which queries can be expressed but the queries should also be expressed in relational calculus to be relationally complete. | For a database language to be relationally complete, the query written in it must be expressible in relational calculus. |
| **8.** | **Query Evaluation** | The evaluation of the query relies on the order specification in which the operations must be performed. | The order of operations does not matter in relational calculus for the evaluation of queries. |
| **9.** | **Database access** | For accessing the database, relational algebra provides a solution in terms of what is required and how to get that information by following a step-by-step description. | For accessing the database, relational calculus provides a solution in terms as simple as what is required and lets the system find the solution for that. |
| **10.** | **Expressiveness** | The expressiveness of any given language is judged using relational algebra operations as a standard. | The completeness of a language is measured in the manner that it is least as powerful as calculus. That implies relation defined using some expression of the calculus is also definable by some other expression, the language is in question. |