Domain Relational Calculus in DBMS

Domain Relational Calculus is a non-procedural query language equivalent in power to Tuple Relational Calculus. Domain Relational Calculus provides only the description of the query but it does not provide the methods to solve it. In Domain Relational Calculus, a query is expressed as,

$\{ < x_1, x_2, x_3, ..., x_n > | P (x_1, x_2, x_3, ..., x_n) \}$

where, $< x_1, x_2, x_3, \ldots, x_n >$ represents resulting domains variables and $P (x_1, x_2, x_3, \ldots, x_n)$ represents the condition or formula equivalent to the Predicate calculus.

Predicate Calculus Formula:

1. Set of all comparison operators
2. Set of connectives like and, or, not
3. Set of quantifiers

Example:

Table-1: Customer

Customer name	Street	City
Debomit	Kadamtala	Alipurduar
Sayantan	Udaypur	Balurghat
Soumya	Nutanchati	Bankura
Ritu	Juhu	Mumbai

Table-2: Loan

Loan number	Branch name	Amount

Loan number	Branch name	Amount
L01	Main	200
L03	Main	150
L10	Sub	90
L08	Main	60

Table-3: Borrower

Customer name	Loan number
Ritu	L01
Debomit	L08
Soumya	L03

Query-1: Find the loan number, branch, amount of loans of greater than or equal to 100 amount.

$\{\prec l, b, a\succ \mid \prec l, b, a\succ \in loan \wedge (a \geq 100)\}$

Resulting relation:

Loan number	Branch name	Amount
L01	Main	200
L03	Main	150

Query-2: Find the loan number for each loan of an amount greater or equal to 150.

$\{\prec l\succ \mid \exists\ b, a\ (\prec l, b, a\succ \in loan \wedge (a \geq 150)\}$

Resulting relation:

Loan number
L01
L03

Query-3: Find the names of all customers having a loan at the “Main” branch and find the loan amount .

{≺c, a≻ | ∃ l (≺c, l≻ ∈ borrower ∧ ∃ b (≺l, b, a≻ ∈ loan ∧ (b = “Main”)))}

Resulting relation:

Customer Name	Amount
Ritu	200
Debomit	60
Soumya	150

Note:
The domain variables those will be in resulting relation must appear before | within ≺ and ≻ and all the domain variables must appear in which order they are in original relation or table.