**Database Management System (KCS-501)**

**Computer Science & Engineering**

**Year/Semester- 3rd /5th**

**Session- 2020-21**

**Unit-2 [Notes-2/10]**

****

**Himanshu Srivastava**

**Department of Computer Science**

**United College of Engineering & Management**

**Relational Algebra**  
The relational algebra is a procedural query language. It consists of a set of operations that take one or two relations as input and produces a new relation as output.

**Fundamental Operations**

* SELECT
* PROJECT
* UNION
* SET DIFFERENCE
* CARTESIAN PRODUCT
* RENAME

**Select, project and rename** - **unary operation** as they operate on a single relation.

**Union, set difference, and Cartesian product** -**binary operations** as they operate on pairs of relations.

**Other Operations**

* SET INTERSECTION
* NATURAL JOIN
* DIVISION
* ASSIGNMENT

**The SELECT Operation**: to identify a set of tuples which is a part of a relation. The select operation selects tuples that satisfy a given predicate or condition.

* It is a unary operation defined on a single relation.
* It is denoted as **σ**.

Consider the following table "Book" :-

+--------+--------+------------------+

| Acc-no | Yr-pub | title |

+--------+--------+------------------+

| 734216 | 1982 | Algorithm design |

| 237235 | 1995 | Database systems |

| 631523 | 1992 | Compiler design |

| 543211 | 1991 | Programming |

| 376112 | 1992 | Machine design |

+--------+--------+------------------+

Example1:- Select from the relation “Book” all the books whose year of publication is 1992.

σYr-pub=1992(Book)

Example2:- Select from the relation “Book” all the books whose Acc-no is greater than equal to 56782.

σ Acc-no>=56782(Book)

**PROJECT Operation**: - returns its argument relation with certain attributes left out.

* It is a unary operation defined on a single relation
* It is denoted as **Π**.

Example:- List all the Title and Acc-no of the “Book” relation.

Π Acc-no, Title(Book)

**The union operation:** - is used when we need some attributes that appear in either or both of the two relations.

* It is denoted as **U**.

Example:  
Borrower (customer-name, loan-number)  
Depositor (customer-name, account-number)  
Customer (customer-name, street-number, customer-city)  
  
List all the customers who have either an account or a loan or both

Π customer-name(Borrower) U Π customer-name (Depositor)

For a union operation **r U s** to be valid, two conditions must hold:

* The relation r and s must be of the same arity, i.e. they must have the same number of attributes.
* The domains of the ith attribute of r and the ith attribute of s must be the same for all i.

**The set difference operation: -**finds tuples in one relation but not in other.

* It is denoted as **–**

Example:Find the names of all customers who have an account but not a loan.

Code:Π customer-name (Depositor)- Π customer-name(Borrower)

**The Cartesian product operation: -**allows combining information from two relations.

* It is denoted as **r X s**where r and s are relations.

Consider the following relation or table "r" :-

+--------+-------+

| A | B |

+--------+-------+

| a | 1 |

| b | 2 |

| a | 2 |

+--------+-------+

Consider another relation or table "s" :-

+--------+-------+

| B | C |

+--------+-------+

| 3 | 1a |

| 2 | 2b |

+--------+-------+

Therefore, rXs gives:-

+-----------+---------+---------+----------+

| r.A |r.B | s.B | s.C |

+-----------+---------+---------+----------+

| a | 1 | 3 | 1a |

| a | 1 | 2 | 2b |

| b | 2 | 3 | 1a |

| b | 2 | 2 | 2b |

| a | 2 | 3 | 1a |

| a | 2 | 2 | 2b |

+-----------+---------+---------+----------+

If relation r has n1 tuples and relation s has n2 tuples then **r** **X s**has n1\*n2 tuples.  
Example:  
Borrower (customer-name, loan-number)  
Loan (loan-number, branch-name, city, amount)  
  
List the names of all customers who have a loan in “Perryridge” branch

Π customer-name (σ Borrower.loan-number=Loan.loan-number (σ branch-name=”Perryridge”(Borrower X Loan)))

**References:-**

* ***Database Systems 7th EDITION***by**RamezElmasri ,Shamkant B. Navathe.**
* **DATABASE SYSTEM CONCEPTS** 6th EDITION by **Abraham Silberschatz , Henry F. Korth ,S. Sudarshan.**