MIS 301 RELATIONAL DATABASE MANAGEMENT SYSTEM

DATABASE MANAGEMENT SYSTEM

The Relational Model And Relational Algebra

LECTURE 6&7

RELATIONAL ALGEBRA

- Relational Algebra is used with relational databases.
- Relational algebra is a procedural query language, which takes instances of relations as input and yields instances of relations as output.
- It uses operators to perform queries.
- Operators used can be either **unary** or **binary**.
- Relational algebra is performed recursively on a relation and intermediate results are also considered relations.
- The fundamental operations of relational algebra are ***Select, Project, Union, Set different, Cartesian product*** and ***Rename***

Relational Algebra-Select Operation

σ

- It selects tuples that satisfy the given logical condition from a relation.
- **Notation** − $\sigma_{<logical\ condition>}$(tablename)
- All tuples satisfying the given condition are extracted from the given table.
- Example: $\sigma_{city="Durgapur"}$(student)→will return all tuples with city = Durgapur from relation student

  $\sigma_{city="Durgapur"\ and\ marks<33}$(student)→will return records of all Durgapur resident students with less than 33 marks

RELATIONAL ALGEBRA-PROJECT OPERATION
Π

- It picks up selected attributes from a relation/table.
- **Notation** − $\Pi_{<a1,a2,\ldots.an>}$(tablename) where a1,a2…..an are attributes.
- Named attributes from all relations are extracted.
- Example: $\Pi_{rollno,\ name,\ city}$ (student) →will return the columns rollno, name and city from all tuples from relation student

$\Pi_{rollno,\ name}\ \sigma_{city=\text{“Durgapur” and marks<33}}$(student)→will return roll number and names of all Durgapur resident students with less than 33 marks

RELATIONAL ALGEBRA-UNION OPERATION U

- It performs binary union between two given relations
- **Notation** − R1**U**R2 is defined as {e|e **Є** R1 or e **Є** R2}
- For a union operation to be valid, the following conditions must hold −
    1. R1 and R2 must have the same number of attributes.
    2. Attribute domains must be compatible.
    3. Duplicate tuples are automatically eliminated.
- Example: $\prod_{city}$ (student) **U** $\prod_{city}$ (teacher) →will give the names of all cities to which either teachers or students or both belong.

RELATIONAL ALGEBRA-INTERSECT OPERATION ∩

- It performs binary intersection between two given relations
- **Notation** − R1 **∩** R2 is defined as {e|e **Є** R1 and e **Є** R2}
- For an intersect operation to be valid, the following conditions must hold −
    1. R1 and R2 must have the same number of attributes.
    2. Attribute domains must be compatible.
    3. Duplicate tuples are automatically eliminated.
- Example: $\prod_{city}$ (student) **∩** $\prod_{city}$ (teacher) →will give the names of all cities to which both teachers and students belong.

Relational Algebra-Difference Operation -

- It performs binary minus between two given relations
- **Notation** − R1 **-** R2 is defined as {e|e **Є** R1 but e not **Є** R2}
- For *a minus/difference* operation to be valid, the following conditions must hold −
    1. R1 and R2 must have the same number of attributes.
    2. Attribute domains must be compatible.
    3. Duplicate tuples are automatically eliminated.
- Example: $\prod_{city}$ (student) **-** $\prod_{city}$ (teacher) →will give the names of all cities to which students belong but teachers do not.
- $\prod_{city}$ (student) **-** $\prod_{city}$ (teacher) is not equal to $\prod_{city}$ (teacher) **-** $\prod_{city}$ (student) unlike the union and intersect operations.

RELATIONAL ALGEBRA-CARTISIAN PRODUCT
X

- It Combines information of two different relations into one.
- **Notation** − R1 **X** R2
- Every tuple of R1 is joined with every tuple of R2
- Example: student **X** teacher→will join every student record with every teacher record
- Number of attributes is sum of attributes of the two relations
- Number of tuples will be product of tuples in the two relations.

Relational Algebra-Rename Operation

ρ

- The results of relational algebra are also relations but without any name.
- The rename operation allows us to rename the output relation.
- **Notation** − ρ_{x} (E) where the result of expression **E** is saved with name of **x**.
- **Example :** ρ_{CITIES} ($\prod_{city}$ (student) **∩** $\prod_{city}$ (teacher)) will store the city names of cities to which teachers as well as students belong in a relation called CITIES.

- TILL WE MEET AGAIN IN THE NEXT CLASS.....