Class room log-2

Tummala Armitha

IMT2014060

Relational Constraints

- Domain Constraints
- Key Constraints
- Referential Integrity Constraints
- Entity Integrity Constraints
- Semantic Integrity Constraints

Constraints.

4.Entity Integrity Constraints:

- Primary Key cannot be null.

5.Semantic Integrity Constraints:

- These are specific to the application and not to database design.
- Example: The minimum age of an employee is 18 and maximum is 60.

Relational Operations

- They are of two types

  1.Updates

  2.Retrieval

Updates - Insert , Delete,Update.

Retrieval – Select, Project,Cartesian join,Theta join

Insert

- Adding a new tuple.
- Domain, key, Referential, Entity constraints can be violated while inserting a new tuple.

Delete

- Removing a tuple.
- Referential integrity may be violated while deleting a tuple.
- Actions taken by the system:

1. Rejecting the deletion.
2. Cascading the effects.
3. Set the value to be null or to default value.

Update

- Changing the existing tuple.
- Works on attributes
- Non-primary, Non-foreign key – Domain constraint is violated.
- Primary key - Can be viewed as deleting and inserting a tuple.
- Foreign key – Referential integrity is violated.

Students

Students_ID	Students_Name	Club	BirthYear
1	Pavithra	C1	1998
2	Vineet	C2	1996
3	Likhitha	C3	1996
4	Anup	C4	1997
5	Shashank	C3	1996

Club

Club_ID	Club_Name	Managed by
C1	Dance	1
C2	Isoc	2
C3	Aikyam	3
C4	Movie	4

Select

- selects a subset of tuples from a relation based on the given condition.
- The syntax is

$$\sigma_{<condition>} (Relation)$$

Example: $\sigma_{<birthyear==1996>} (students)$

Select

- The output is:

Students_ID	Students_name	Club	BirthYear
2	Vineet	C2	1996
3	Likhitha	C3	1996
5	Shashank	C3	1996

Properties of Select

- Unary
- Degree of relation is same as that of input.
- $|\sigma_c(R)| <= |R|$
- Commutative --- $\sigma_{C1}(\sigma_{c2}(R)) = \sigma_{c2}(\sigma_{c1}(R))$

project

- Gives the columns that satisfy the given predicate.
- The syntax is

$\prod_{\text{<attributes>}} (\text{Relation})$

Examples: $\prod_{\text{<Club>}} (\text{Students})$

Project

- The output is:

Club
C1
C2
C3
C4

Properties of Project

- Unary
- Eliminates Duplication
- $| \prod_{at} (R) | <= |R|$
- Degree == number of attributes given
- Not Commutative.

Composition & Assignment

- Output and input of operators is a single relation, so they can be composed in any fashion.

Example:$\prod_{<name>} (\sigma_{< birthyear>1996 >} (students))$

Composition & Assignment

- Results of a query can also be assigned to a name to form a relation by that name
- Example : studentsname $\leftarrow \prod_{<name>} (\sigma_{< birthyear>1996 >}(students))$

Question

What if we want both students name and Club name ?

select and project works on single relation, so we cannot use them.

Cartesian join

- Join of every row of one table to every row of another table.
- Makes two tables into one
- If there are m tuples in table 1 and n tuples in table 2 , then there are m * n tuples in resultant table.
- The Cartesian join is costly.

Cartesian join

Students_ID	Students_name	club	Birthyear	Club_ID	Club name	Managed by
1	Pavithra	C1	1998	C1	Dance	1
1	Pavithra	C1	1998	C2	Isoc	2
1	Pavithra	C1	1998	C3	Aikyam	3
1	Pavithra	C1	1998	C4	Movie	4
2	Vineet	C2	1996	C1	Dance	1
2	Vineet	C2	1996	C2	Isoc	2
2	Vineet	C2	1996	C3	Aikyam	3

Cartesian join

- The output of $\prod_{\text{<Students.name,Club.name>}}(\sigma_{\text{<Students.Club = Club.Club\_ID >}}(\text{students X Club }))$

Students_name	Club_name
Pavithra	Dance
Vineet	Isoc
Likhitha	Aikyam
Anup	Movie
Shashank	Aikyam

Theta join(θ)

- combine related tuples from two or more relations (specified by the condition θ), to form a single tuple.
- Output is same as that of Cartesian join
- The syntax is

$$A \bowtie_{\theta} B$$

Theta join(θ)

$\text{Students} \bowtie_{\theta= \text{Students.Club = Club.Club\_ID}} \text{Club}$

Students_ID	Students_name	Club	BirthYear	Club_ID	Club_name	Managed by
1	Pavithra	C1	1998	C1	Dance	1
2	vineet	C2	1996	C2	Isoc	2
3	Likhitha	C3	1996	C3	Aikyam	3
4	Anup	C4	1997	C4	Movie	4
5	Shashank	C3	1996	C3	Aikyam	3

Equi and Natural Join

- Equi join – Theta join with conditional operator = .
- Natural join(*) – Performs Equi join only on same attributes.

Natural join

Students		
ID	Name	Department
1	A	D1
2	B	D2
3	C	D3

Departments	
Department	Dept_name
D1	CS
D2	EEE
D3	Mechanical

Natural join

ID	Name	Department	Dept_name
1	A	D1	CS
2	B	D2	EEE
3	C	D3	Mechanical

Set Operations

- Union
- Intersection
- Difference

Set Operations

- Conditions to be satisfied to apply set operations.

  1.Same number of attributes.

  2.Domain of attributes should be same and in same order.

Union

Table1	
ID	Name
1001	Adam
1002	Abhi

Table2	
ID	Name
1002	Abhi
1003	chester

Union

The union of Table 1 and Table2:

ID	Name
1001	Adam
1002	Abhi
1003	Chester

Intersection & Difference

- The intersection and Difference of Table1 and Table2 :

Intersection	
ID	Name
1002	Abhi

Difference	
ID	Name
1001	Adam

Thank you