# Computer Science & IT

## Database Management System

Entity Relationship Model
&
Integrity constraints

Lecture No. 01

By- Vishal Sir

# Recap of Previous Lecture

**Topic** — Deletion from B+ tree

**Topic** — Practice questions

Slide

# Topics to be Covered

**Topic** — Relational model

**Topic** — Integrity constraints

Slide

## Topic : Relational Model concept

☑ The relational model was proposed by E.F. Codd to model data in the form of relations or tables.

☑ The relational model is represented by a table with columns and rows. Each row is known as a tuple.

✓ These are the restrictions or set of rules imposed on the database content. It validates the quality of the database. It validates that various operations like data insertion, updation, etc., can be performed without affecting the integrity of the data.

✓ Constraints could be either on a column level or a table level. The column level constraints are applied only to one column, whereas the table level constraints are applied to the whole table.

1. Domain constraint

2. Key constraint or Tuple Uniqueness constraint

3. Entity Integrity constraint

4. Referential Integrity constraint

→ Every domain must contain atomic values(smallest indivisible units) which means composite and multi-valued attributes are not allowed.

• We perform a datatype check here, which means when we assign a data type to a column we limit the values that it can contain. Eg. If we assign the datatype of attribute age as int, we can't give it values other than int datatype

These are called ^tuple uniqueness constraints since it ensures that every tuple in the relation should be unique. { i.e, we must be able to define the Candidate key }

A relation can have multiple keys or candidate keys(minimal superkey), out of which we choose one of the keys as the primary key, we don't have any restriction on choosing the primary key out of candidate keys.

→ Entity integrity constraint specifies that no attribute of primary key must contain a null value in any relation.

The Referential integrity constraint is specified between two relations or tables and used to maintain the consistency among the tuples in two relations.

Some times may be on the same table

➤ **Foreign Key:** A foreign key is a set of attributes in a table that refers to the primary key or alternate key of same table or some other table.

eg:

Foreign Key ↓ → It references to the Stu-id attribute of Student table

Primary Key

Primary Key

Names of the attributes need not be same

| Sid | Cid |
|-----|-----|
| S1 | C1 |
| S1 | C2 |
| S3 | C2 |
| S4 | C3 |

Enroll

It is also a Foreign key that references to the attribute 'Cid' of Course table

| Stu-id | Sname | Branch |
|--------|-------|--------|
| S1 | | |
| S2 | | |
| S3 | | |
| S4 | | |

Student

| Cid | Cname | Branch |
|-----|-------|--------|
| C1 | | |
| C2 | | |
| C3 | | |
| C4 | | |

Course

{ Enroll is the referencing relation }

{ Student is the referenced relation }

Note:
① ✓ A relation that contains foreign key is called a referencing relation

② ✓ Relation that contains the primary key / alternate key which is being referenced by the foreign key is called a referenced relation

in some cases, referencing relation & referenced relation may be same

eg:-

it references to the
Primary key of same table

Primary
key

Foreign key

| Eid | Ename | . . . . . | Manager_id |
|-----|-------|-----------|------------|
| E1  |       |           | E3         |
| E2  |       |           | E5         |
| E3  |       |           | E6         |
| E4  |       |           | E6         |
| E5  |       |           | NULL       |
| E6  |       |           | E5         |

Employee

Employee relation is
referencing relation
as well as
referenced relation

Foreign key Attribute
May take NULL values

➤ Referential integrity constraint is enforced through a foreign key

Let foreign key in relation R1 refers to primary key of relation R2.

*i.e. $R_1$ is referencing rel$^n$*

*i.e. $R_2$ is referenced relation*

The values of the foreign key in a tuple of relation R1 can either take the values *from*

the primary key *Values of* some tuple in relation R2, or can take NULL values, but can't be

empty.

**Insertion:** Insertion of a tuple in referenced relation does not result in any referential integrity violation.

**Deletion:** Deletion of a tuple from referenced relation may result in referential integrity violation.

**Updation:** Updation of a tuple in referenced relation may result in referential integrity violation.

**Insertion:** Insertion of a tuple in referencing relation may result in referential integrity violation.

**Deletion:** Deletion of a tuple from referencing relation does not result in any referential integrity violation.

**Updation:** Updation of a tuple in referencing relation may result in referential integrity violation.

"W.r.t. Deletion operation"

(or Alternate Key)

Deletion of a primary key value from the referenced relation may causes integrity violation. We may choose one of the following approaches to avoid integrity violation.

✓On Delete No Action:

✓On Delete Cascade:

✓On Delete Set NULL:

**On delete No action**

If deletion of a tuple from referenced relation causes any integrity violation, then that deletion is prohibited.

{ i.e. No action will be performed }
Not even the deletion from referenced relation

## On delete Cascade

* If deletion of a tuple from referenced relation causes any integrity violation, then delete the tuple from the referenced relation, and also delete the corresponding tuples from referencing relation

&rarr; Some times it may result in the deletion of few more tuples

& when referencing & referenced relation are same

If we delete the tuple w.r.t. Eid = "E4" and we have implemented "On delete cascade" then total no. of tuples that will be deleted from the relation Employee is

Primary key

Foreign key

| Eid | Ename | . . . . . | Manager-id |
|-----|-------|-----------|------------|
| E1  |       |           | E3         |
| E2  |       |           | E5         |
| E3  |       |           | E6         |
| E4  |       |           | E6         |
| E5  |       |           | NULL       |
| E6  |       |           | E5         |

Employee

Ans = $\underline{\text{'1'}}$ ?

Only one tuple will be deleted.

If we delete the tuple w.r.t. Eid = E5, and we have implemented "On delete cascade" then total no. of tuples that will be deleted from the relation Employee is

Primary key

Foreign key

| Eid | Ename | . . . . . | Manager-id |
|-----|-------|-----------|------------|
| E1  |       |           | E3         |
| E2  |       |           | E5         |
| E3  |       |           | E6         |
| E4  |       |           | E6         |
| E5  |       |           | NULL       |
| E6  |       |           | E5         |

Employee

Ans = 6

?

All '6' tuples will be deleted.

# On delete Set NULL

\* **Case ①** If deletion of a tuple from referenced relation causes any integrity violation and foreign key attribute is allowed to take the NULL value, then delete the tuple from the referenced relation and Set the value of the foreign key attribute as "NULL" in the tuples that causes integrity violation

--------------------------------

\* **Case ②** If deletion of a tuple from referenced relation causes any integrity violation and foreign key attribute is Not allowed to take the NULL value, then deletion from the referenced relation itself is prohibited, i.e, if foreign key attribute is not allowed to take NULL values then "On delete Set NULL" is Same as "On delete No action"

*w.r.t. Update op^n*

Updation in the primary key value in the referenced relation may causes integrity violation. We may choose one of the following approaches to avoid integrity violation.

On Update No Action:

On Update Cascade:

On Update Set NULL:

## On Update No action

If updation of a tuple in referenced relation causes any integrity violation, then that updation is prohibited.

{ i.e. No action will be performed }
   Not even the updation in referenced relation

# On Update Cascade

If updation of a tuple in referenced relation causes any integrity violation, then update the tuple in the referenced relation, and also update the values in the foreign key column of the referencing relation

# On Update Set NULL

**Case ①** If updation of a tuple in referenced relation causes any integrity violation and foreign key attribute is allowed to take the NULL value, then update the tuple in the referenced relation and set the value of the foreign key attribute as "NULL" in the tuples that causes integrity violation

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

**Case ②** If updation of a tuple in referenced relation causes any integrity violation and foreign key attribute is not allowed to take the NULL value, then updation in the referenced relation itself is prohibited, i.e., if foreign key attribute is not allowed to take NULL values then "On Update Set NULL" is same as "On Update No Action"

**Insertion:** Insertion of a tuple in referencing relation may result in referential integrity violation.

**Deletion:** Deletion of a tuple from referencing relation does not result in any referential integrity violation.

**Updation:** Updation of a tuple in referencing relation may result in referential integrity violation.

Referenced
relation is
Parent relation

Whereas

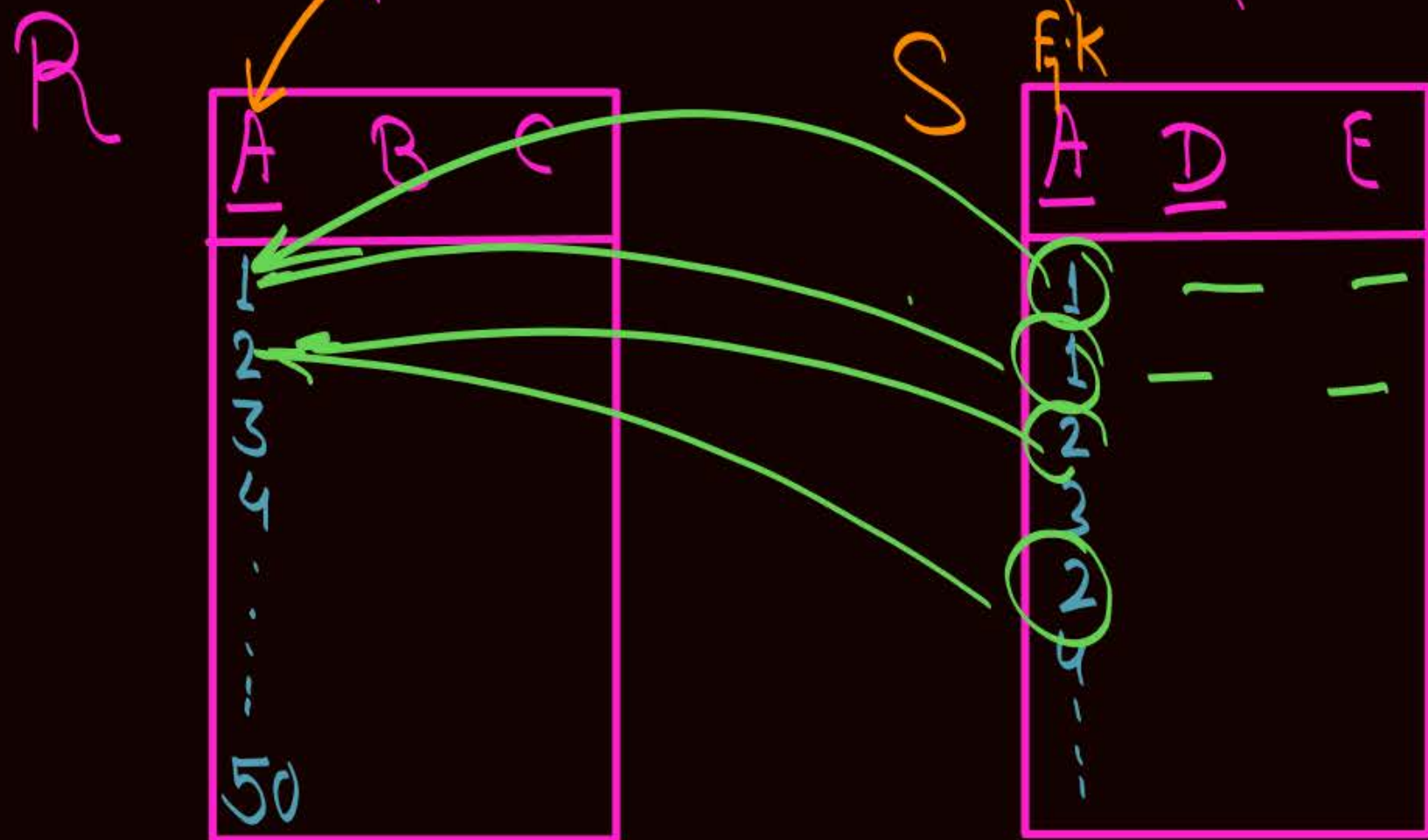Referencing relation
is child relation

If any operation in referencing relation (child relation) causes any referential integrity violation, then corresponding operation is prohibited.

**eg:-** Let R (A B C) is a relation with 50 tuples,
and S (A D E) is a relation with 25 tuples,
then number of tuples in R ⋈ S = ?

Where attribute 'A' in relation S is a foreign key that
references to attribute A of relation R

R

| A | B | C |
|---|---|---|
| 1 | | |
| 2 | | |
| 3 | | |
| 4 | | |
| ... | | |
| 50 | | |

S   F.K

| A | D | E |
|---|---|---|
| 1 | — | — |
| 1 | — | — |
| 2 | | |
| 2 | | |
| 4 | | |

No. of tuples in R ⋈ S = 25

Each tuple of 'S' will match
with exactly '1' tuple of R
∴ in the Result No. of tuples
will be exactly same as
No. of tuples in both 'S'

eg:- Let R(A B C) is a relation with 50 tuples, and S(A D E) is a relation with 25 tuples,

then minimum & maximum no. of tuples possible in R ⋈ S, when 'A' in rel$^n$ 'S' is the F.K, that references to the attribute A of relation R.

R

| A |
|---|
| 1 |
| 2 |
| 3 |
| . |
| . |
| 50 |

| S | A | D | E |
|---|---|---|---|
|  | NULL | 1 | 2 |
|  | NULL | 1 | 3 |
|  | 1 | 2 | 1 |
|  | 2 | 3 | 5 |
|  | . | . | . |
|  | . | . | . |

No of tuples in R ⋈ S =

Minimum = 0
When all values w.r.t Column A in rel$^n$ S are NULL

Maximum = 25
When No values is NULL in Column 'A' of relation S

**Topic** ▸ Relational model

**Topic** ▸ Integrity constraints

Slide