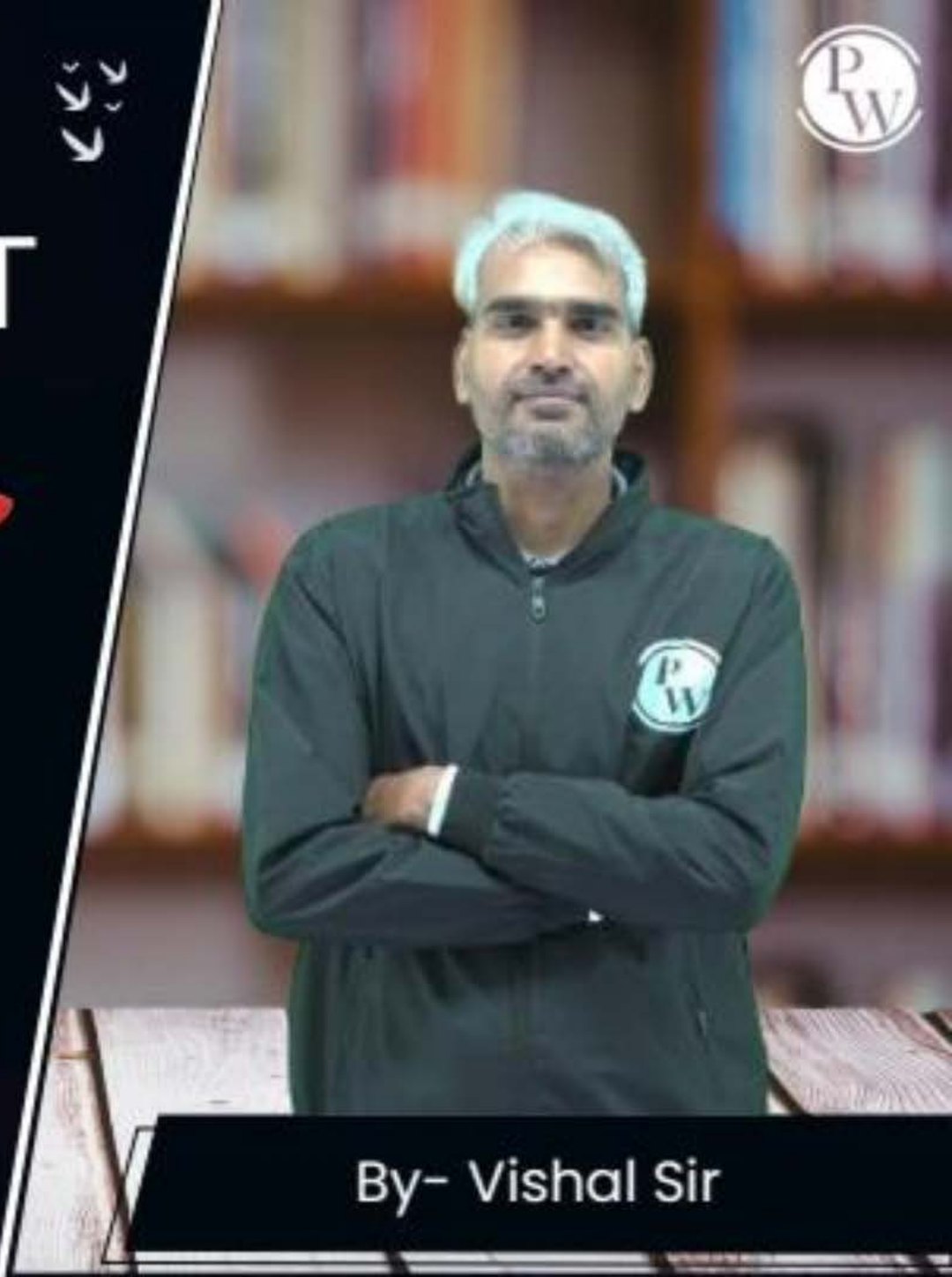
Computer Science & IT

Database Management System

Relational Model & Normal Forms

Lecture No. 02



By- Vishal Sir

Recap of Previous Lecture

- **Topic** Syllabus
- **Topic** Introduction to DBMS
- **Topic** Limitation of file system

Topics to be Covered

- **Topic** Introduction Relational Database Model
- **Topic** Functional dependency

Topic : Relational database

- In relational database information is organized in the form of table {i.e. Collection of rows & Column}
  - Each table of a relational Model is called a relation
  - Dr Codd defined 13 rules {from 0 to 12} for a table to be called a relation.

eg: Consider the following relation "Student"

Student

Name of relation

Attributes / fields

Sid	Sname	Branch
S_1	A	CS
S_2	A	CS
S_3	B	IT
S_4	C	CS

Records / tuples

Each row of relational table is called record / tuple

tuple: (S_4, C, CS)

$\neq (C, S_4, CS)$

Set $\{a, b\} = \{b, a\}$

order pair

$(a, b) \neq (b, a)$

order-triple

(a, b, c)

Quadruple (a, b, c, d)

5-tuple: (a, b, c, d, e)

* **Degree / Arity :** Number of Columns / attributes / fields in a relation define the degree/arity of that relation

* **Cardinality :-** Number of records / tuples in a relational table defines the Cardinality of that relational table

* **Relational Schema :-** Relational schema will provide the abstract details of the relational table, i.e Name of Relation (Name of 1st attribute, Name of 2nd attribute, ------)

* **Relational instance :-** If records / tuples are present in a relation then set of all those tuples is called relational instance of that time

eg: Consider the following relation "Student"

Student

Sid	Sname	Branch
S_1	A	CS
S_2	A	CS
S_3	B	IT
S_4	C	CS

Degree/Arity = 3

Cardinality = 4

Relational Schema
= Student(Sid, Sname, Branch)

Relational instance
= { (S_1, A, CS)
(S_2, A, CS)
(S_4, C, CS)
(S_3, B, IT) }

Topic : Functional dependency (FD)

* Functional dependencies are used to represent the relationship between the sets of attributes of a relation

Let X and Y are two sets of attribute/s w.r.t. relation R, then

$X \longrightarrow Y$ denotes a Functional dependency

↳ FD $X \rightarrow Y$ may or may not hold true in relation R

Topic : Functional dependency (FD)

In functional dependency $X \longrightarrow Y$ { it is read as X determines Y }

X is called determinant

& Y is called dependent

"$X \longrightarrow Y$" represent that if we know X, then we can determine Y. { but Converse need not be true, ie from $X \rightarrow Y$ we can not say that if we know Y then we can determine X }

Topic : Functional dependency (FD)

$R(A_1, A_2, A_3, A_4, A_5)$

- Let R be the relational schema with X and Y as the attribute sets over relation R.

$\{A_1, A_3\}$ $\{A_3, A_5\}$

if (Functional dependency $X \rightarrow Y$ exists in R)
then (For all pair of tuples $t1, t2 \in R$
If $t1.X = t2.X$ then $t1.Y = t2.Y$

if '$X \rightarrow y$' exists in relation R.
then whenever values w.r.t. attributes of set X are repeated
then Corresponding values of set Y will also be repeated

Note: If functional dependency $X \longrightarrow Y$ exists in relation R then,

① Whenever X values are same in two tuples, Corresponding Y values will also be same

② If X values are not same in two tuples, then Y values may or may not be same

③ If 'Y' values are same in two tuples, the 'X' values need not be same in those tuples.

It is necessary Condⁿ for FD $X \rightarrow Y$ to hold true in a relation, but not sufficient

↓

if necessary Condⁿ it-self is false then it is guaranteed that $X \rightarrow Y$ can not hold true in that relation

Topic : Functional dependency (FD)

- If necessary condition for functional dependency "X→Y" does not hold true based on given relation instance, then functional dependency "X→Y" can never exist in the given relation.

- Even if necessary condition for functional dependency "X→Y" does hold true based on given relation instance, then also we can not be sure whether functional dependency X→Y exists in the relation or not, because it is just the relational instance. & it may change with time

- Functional dependencies that hold true in a relation are always identified by database designer based on properties of attribute

  - Functional dependencies that holds true in a relation are given with the question

#Q. From the following instance of a relation schema R (A,B,C), we can conclude that:

A	B	C
1	1	1
1	1	0
2	3	2
2	3	2

Necessary Cond^{n} satisfied for $A \rightarrow B$, but it is not a sufficient cond^{n}

$\therefore$ We can only say that "$A \rightarrow B$" may hold true but we can not Conclude that $A \rightarrow B$ holds

Necessary Cond^{n} for $B \rightarrow C$ is dissatisfied $\therefore$ We can conclude that $B \rightarrow C$ does not hold.

i.e.; we can say with guarantee

(A) A functionally determines B, and B functionally determines C

(B) A functionally determines B, and B does not functionally determine C

(C) B does not functionally determine C

(D) A does not functionally determine B, and B does not functionally determine C

#Q. Consider the following relational instance

A	B	C
1	2	3
1	2	4
2	2	1
3	1	2
4	1	2

Which of the following functional dependency may hold true (not necessarily) based on given relational instance.

a) $A \rightarrow C$

b) $A \rightarrow B$

c) $AB \rightarrow C$

d) $BC \rightarrow A$

#Q. Consider the following relational instance

A	B	C
1	1	1
1	2	2
2	4	3
3	3	4
4	1	5
5	3	6

Which of the following functional dependency may hold true (not necessarily) based on given relational instance.

a) $B \rightarrow C$

b) $A \rightarrow B$

c) $C \rightarrow B$

d) $B \rightarrow A$

e) $C \rightarrow A$

Topic : Types of Functional Dependency

There are two types of Functional dependencies

- Trivial Functional dependency
- Non-trivial functional dependency

We may define a third type of functional dependency

i.e. "Semi-non-trivial FD"

└ It is mix of trivial & non-trivial

Topic : Trivial Functional Dependency

Let X and Y are two sets of attributes from relation R.

FD $X \longrightarrow Y$ is called trivial if and only if $X \supseteq Y$

eg: Sid $\longrightarrow$ Sid
Sid, Sname $\longrightarrow$ Sname
Sid, Sname $\longrightarrow$ Sid

} All are trivial FDs

+ Every trivial functional dependency possible with the attributes of relation R will always hold true in the relation R

Topic : Non-trivial Functional Dependency

Let X and Y are two non-Empty sets of attributes from attributes of relation R.

Functional dependency $X \longrightarrow Y$ is called Non-trivial FD if and only if

$$X \cap Y = \emptyset$$

e.g.
- Sid $\longrightarrow$ Sname
- Sid, Sname $\longrightarrow$ Branch
- Sname $\longrightarrow$ Branch

} All are examples of non-trivial FDs.

- Non-trivial functional dependencies that can be defined using the attributes of relation R, need not hold true in the relation R

eg: Consider the following relational schema
Student (Sid, Sname, Branch)
Write all possible non-trivial functional dependencies w.r.t. attributes of relation "Student"

Sid → Sname
Sid → Branch
Sname → Sid
Sname → Branch
Branch → Sid
Branch → Sname

Sid → Sname, Branch
Sname → Sid, Branch
Branch → Sid, Sname

Sid, Sname → Branch
Sid, Branch → Sname
Sname, Branch → Sid

These are the only non-trivial functional dependencies that can be defined w.r.t. attributes of relation Student {It is not necessary for all of them to hold true in the relation Student}

Topic : Semi non-trivial Functional Dependency

Let X & Y are two non-empty sets of attributes from the attributes of relation R.

Functional dependency $X \longrightarrow Y$ is called Semi-non-trivial FD if and only if (i) $X \not\supseteq Y$

and (ii) $X \cap Y \neq \phi$

e.g. Sid, Sname $\longrightarrow$ Sname, Branch } Semi-non-trivial FD

* We will be able to split semi non-trivial FD into two parts, i.e., trivial part and non-trivial part,
  ∴ We don't need to discuss the semi non-trivial FDs seperately.

H.W.

#Q. Consider the following relational instance

A	B	C
1	1	1
1	1	2
2	1	2
2	2	3
3	3	4

Find all non-trivial FDs which may hold true in the above relation based on given relational instance.

GATE 2024

FD $X \rightarrow Y$ is called an useful FD if and only if both X and Y are non-empty sets and

$$X \cap Y = \phi$$

it is the definition of non-trivial FD

How many useful FDs are possible in a relation with '4' attributes ?

Q:- How many non-trivial FDs are possible in a relation with "4" attributes

(Same as previous question)

Q: Let R(A B C D E) be the relational schema.
How many non-trivial FDs can be defined in relation R

2 mins Summary

Topic Introduction Relational Database Model

Topic Functional dependency

Slide

THANK - YOU