

# Computer Science & IT

## Database Management System



Relational Model & Normal Forms

Lecture No. 03



By- Vishal Sir

# Recap of Previous Lecture



Topic

Introduction Relational Database Model



Topic

Functional dependency



# Topics to be Covered



Topic

Properties of functional dependency



Topic

Different types of keys in RDBMS



Topic

Candidate key



Topic

Super key



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

Possible non-trivial FDs w.r.t. three attributes 'A', 'B' & 'C'

~~$A \rightarrow B$~~

~~$A \rightarrow BC$~~

~~$AB \rightarrow C$~~

~~$A \rightarrow C$~~

~~$B \rightarrow AC$~~

$AC \rightarrow B$

~~$B \rightarrow A$~~

~~$C \rightarrow AB$~~

~~$BC \rightarrow A$~~

~~$B \rightarrow C$~~

~~$C \rightarrow A$~~

$C \rightarrow B$

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

Ans:  $C \rightarrow B$ ,  $AC \rightarrow B$

GATE 2024

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

$$X \cap Y = \emptyset$$

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 let attributes are A, B, C & D

(Same as previous question)

$A \rightarrow B$	$C \rightarrow A$	$A \rightarrow BC$	$C \rightarrow AB$	$A \rightarrow BCD$
$A \rightarrow C$	$C \rightarrow B$	$A \rightarrow BD$	$C \rightarrow AD$	$B \rightarrow ACD$
$A \rightarrow D$	$C \rightarrow D$	$A \rightarrow CD$	$C \rightarrow BD$	$C \rightarrow ABD$
$B \rightarrow A$	$D \rightarrow A$	$B \rightarrow AC$	$D \rightarrow AB$	$D \rightarrow ABC$
$B \rightarrow C$	$D \rightarrow B$	$B \rightarrow AD$	$D \rightarrow AC$	
$B \rightarrow D$	$D \rightarrow C$	$B \rightarrow CD$	$D \rightarrow BC$	

28 FDs

$AB \rightarrow C$	$AB \rightarrow CD$
$AB \rightarrow D$	$AC \rightarrow BD$
$AC \rightarrow B$	$AD \rightarrow BC$
$AC \rightarrow D$	$BC \rightarrow AD$
$AD \rightarrow B$	$BD \rightarrow AC$
$AD \rightarrow C$	$CD \rightarrow AB$
$BC \rightarrow A$	
$BC \rightarrow D$	
$BD \rightarrow A$	
$BD \rightarrow C$	
$CD \rightarrow A$	
$CD \rightarrow B$	

18 FDs

$ABC \rightarrow D$
$ABD \rightarrow C$
$ACD \rightarrow B$
$BCD \rightarrow A$

4 FDs

$$\text{Total no. of FDs} = 28 + 18 + 4 = 50$$

Q:-  
↓  
(Same as previous question)

How many non-trivial FDs are possible in a relation with "4" attributes (let attributes are A, B, C, & D)

Relation contains 4 attributes, ∴ In any non-trivial FD  $X \rightarrow Y$  we know  $1 \leq |X| \leq 3$

Case ① When  $|X| = 1$ , then  $|Y| = 1$  or 2 or 3

$$X \longrightarrow Y$$

$$4C_1 \text{ and } (3C_1 \text{ or } 3C_2 \text{ or } 3C_3) = 4C_1 * (3C_1 + 3C_2 + 3C_3) = 4 * (3 + 3 + 3) = 36$$

Case ② When  $|X| = 2$ , then  $|Y| = 1$  or 2

$$X \longrightarrow Y$$

$$4C_2 * (2C_1 + 2C_2) = 6 * (2 + 1) = 6 * 3 = 18$$

Case ③ When  $|X| = 3$ , then  $|Y| = 1$

$$X \longrightarrow Y$$

$$4C_3 * (1C_1) = 4 * 1 = 4$$

Total no.

$$\text{of FDs} = 36 + 18 + 4 = 58$$

Q: Let  $R(AB C D E)$  be the relational schema.  
How many non-trivial FDs Can be defined in relation R  
w.r.t. non-trivial FD  $X \rightarrow Y$

Case ① When  $|X|=1$   $\{ {}^5C_1 * (2^4 - 1) \} = 5 * 15 = 75$

Case ② When  $|X|=2$   $\{ {}^5C_2 * (2^3 - 1) \} = 10 * 7 = 70$

Case ③ When  $|X|=3$   $\{ {}^5C_3 * (2^2 - 1) \} = 10 * 3 = 30$

Case ④ When  $|X|=4$   $\{ {}^5C_4 * (2^1 - 1) \} = 5 * 1 = 5$

---

Total no. of FDs =  $75 + 70 + 30 + 5 = 180$



## Topic : Properties of Functional Dependencies

- ① Reflexivity  $\Rightarrow$  FD  $X \rightarrow Y$  is called a reflexive FD only if  $X \supseteq Y$   
Every Reflexive FD will always hold true in relation
- ② Augmentation  $\Rightarrow$  Let FD  $X \rightarrow Y$  exist in relation R  
then  $XZ \rightarrow YZ$  will also hold true in relation R
- ③ Transitivity  $\Rightarrow$  Let  $X \rightarrow Y$  and  $Y \rightarrow Z$  holds true in relation R  
then  $X \rightarrow Z$  will also hold true in relation R

These three properties are called "Armstrong's Axioms"



## Topic : Properties of Functional Dependencies

### ④ Decomposition : (Splitting Rule)

If FD  $X \rightarrow YZ$  holds true in relation R  
then  $X \rightarrow Y$  &  $X \rightarrow Z$  will also hold  
true in relation R

If  $XY \rightarrow Z$  exists  
then  $X \rightarrow Z$  and  
 $Y \rightarrow Z$  need not  
hold true in the relation

We can not split  
the set at "L.H.S."

Let  $\overbrace{AB \rightarrow BC}^{\text{semi-non-trivial FD}}$  holds true in relation  
then By splitting rule.  
 $AB \rightarrow B$  { Trivial Part of  $AB \rightarrow BC$  }  
&  $AB \rightarrow C$  { Non-trivial part of  $AB \rightarrow BC$  }  
We can always decompose semi non-trivial FDs  
in to two parts (i) Trivial part & (ii) Non-trivial part



## Topic : Properties of Functional Dependencies

⑤ Union:- If  $X \rightarrow Y$  and  $X \rightarrow Z$  holds true in the relation then " $X \rightarrow YZ$ " will also hold true in relation

⑥ Composition:- If  $X \rightarrow Y$  and  $P \rightarrow Q$  holds true in relation then  $XP \rightarrow YQ$  will also hold true in relation

⑦ Pseudo Transitivity:- If  $X \rightarrow Y$  and  $YW \rightarrow Z$  holds true in  $rel^h$  then  $XW \rightarrow Z$  will also hold true in relation

# Key Concept :-

★ In a relational table no two tuples should be exactly same

★  $\rightarrow$  i.e., In a relational table duplicate tuples are not allowed

★  $\rightarrow$  To implement this restriction every relation must have a key.

**"Key"**  $\rightarrow$  A key in a relation is the set of attributes that can uniquely identify each tuple in the relation



## Topic : Different types of keys

There are various types of Keys

- ✓ ① Candidate Key
- ✓ ② Primary Key
- ✓ ③ Alternate Key (Secondary Key)
- ✓ ④ Super Key

⑤ Foreign Key

← "Foreign Key" is not actually a Key  
\* We will discuss about foreign key during the discussion of ER Model

\* Minimal Set  $\rightarrow$  { A set of elements from which no element can be removed without losing the associated property is called a minimal set.

{ i.e., Whenever we remove any element from the set then it is guaranteed that it will lose the associated property.

Note:-→ If values of a set of attributes are guaranteed to be unique in a relation, then that set of attributes is definitely a "key" of that relation. { But it may or may not be minimal }



## Topic : Candidate key

A minimal set of attributes that can uniquely identify each tuple of the relation is called a Candidate Key

★ ie, A set of attributes from which no attribute can be removed without destroying its property of being a key is called a Candidate Key.

Another definition:- A minimal set of attributes that can determine all attributes of the relation is called a Candidate Key.

eg①:- Consider the following relation

Student

Sid	Sname	Fee
S <sub>1</sub>	A	500
S <sub>2</sub>	A	500
S <sub>3</sub>	B	600
S <sub>4</sub>	B	400
S <sub>5</sub>	C	600

Also consider that following functional dependencies holds in the relation

$Sid \rightarrow Sname$   
 $Sid \rightarrow fee$   
 $Sid \rightarrow Sid$  holds

We know

Augmentation  
at  
Sname

$\{Sid, Sname\} \rightarrow \{Sid, Sname, fee\}$   
 all attributes

$\therefore (Sid, Sname)$  is also a key  
 but it is not minimal, because we can remove Sname from the set

$Sid$  is a minimal key  
 $\therefore Sid$  is a Candidate key

Union

$Sid \rightarrow Sid, Sname, fee$   
 all attributes

$\therefore Sid$  is a key  
 (also minimal)

eg②: Consider the following relation

Enroll

Sid	Cid	I-id
S <sub>1</sub>	C <sub>1</sub>	101
S <sub>2</sub>	C <sub>1</sub>	101
S <sub>3</sub>	C <sub>2</sub>	101
S <sub>3</sub>	C <sub>3</sub>	102
S <sub>4</sub>	C <sub>3</sub>	102

Consider following functional dependencies holds in the relation

$Cid \rightarrow I-id$

$Cid \rightarrow Cid, I-id$

Augment Sid

$\{Sid, Cid\} \rightarrow \{Sid, Cid, I-id\}$

all attributes

$\therefore \{Sid, Cid\}$  is a key

If we remove "Cid" from the set, then "Sid" alone can not uniquely identify each tuple of reln.  
 i.e. Sid is not a key

Similarly if we remove "Sid" from the set then Cid alone can not uniquely identify each tuple of relation  
 i.e. "Cid" is not a key

i.e. we can not remove any attribute from set  $\{Sid, Cid\}$  without losing its property of being a key  
 Hence  $\{Sid, Cid\}$  is minimal key  
 i.e. a Candidate key



## Topic : Candidate key

- ① A key with a single attribute is always minimal, hence a key with a single attribute is always a Candidate Key.
- ② If a Candidate key is formed of a single attribute, then it is called a simple Candidate Key.
- ③ If a Candidate key is formed of two or more attributes, then it is called Compound or Composite Candidate key.



## Topic : Candidate key

- ④ A relation may have more than one candidate key.  
{ Number of Candidate keys in a relation will be identified by functional dependencies that holds true in the given relation }
- ⑤ Attribute that belongs to any of the candidate key is called "Prime attribute" (or key attribute)
- ⑥ Attribute that does not belong to any candidate key of relation is called "non-prime attribute" (or non-key attribute)



## 2 mins Summary



**Topic**

Properties of functional dependency

**Topic**

Different types of keys in RDBMS

**Topic**

Candidate key

**Topic**

Super key

telegram t.me// vishalRawtiya PW



**THANK - YOU**