

Normalizing Database using function dependencies upto Normal form.

Aim: To normalize the below relation and create the simplified table with suitable constraint.

Cricket Board (Board ID, Name, Address, contact No, Team ID, T Name, coach, captain, player ID, PF Name, PL Name, Age, Date of Birth, playing role, email, contact - No, Batting, Bowling, match ID, match - Date, Time, result, Ground ID, G Name, location, capacity, umpire ID, UF Name, UL Name, UAge, UDate of Birth, country, uemail, Ucontact - no)

Procedure:

Normalize the given relation and create simplified tables with suitable constraints, we need to identify the functional dependencies and separate them into different tables. Normalization involves breaking down the data redundancy and maintain data integrity. Let's identify the functional dependencies.

Functional dependency:

Now we can create simplified tables:

Cricket Board (Board ID [PK], Name, Address, contact - No)

Cricket Team (Team ID [PK], T Name, coach, captain)

Cricket player (player ID [PK], Team ID [FK],

Cricket match (match ID [PK], Team ID [FK] match - Date

Time, result, Ground ID [FK])

In these tables, [PK] denotes the Primary key [FK] denotes the foreign key, and suitable constraints should be added to maintain data integrity.

Create tables for all non-prime attributes based on their functional dependencies and candidate keys. And check for each set of attributes that functionally depend on a candidate key. The candidate keys in this case are Board ID, Team ID, Player ID, match ID, and umpire.

First Normal Form:

The given relation into the first normal form (1NF). To ensure that each attribute contains atomic values, and there are no repeating groups or arrays. Based on the provided relation, it appears that each attribute already contains atomic values, so there are no repeating groups to eliminate.

Second Normal Form:

To determine whether the given relation is in the second normal form, we need to check 2 conditions. The relation must be already in 1NF. It appears that potential candidate keys could be:

1. Board ID
2. Team ID
3. Player ID
4. match ID
5. umpire ID.

Third Normal Form:

we need to check 2 conditions.

1. The relation must be in 2NF.
2. There should be no transitive dependencies between non prime attributes and candidate key.

Board ID \rightarrow name, Address, contact - NO.

There are no transitive dependencies in this case, as name, Address, and contact - NO are directly dependent on Board ID.

Player ID → PF Name, PL Name, Age, PDOB, player role, email, contact_no, Batting, Bowling.

There are no transitive dependencies for Ground ID as GName, Location, and Capacity are directly dependent on Ground ID.

Umpire ID → UF Name, UL Name, UAge, UDOB, country, Uemail, Ucontact_no.

VEL TECH - CSE	
EX NO.	
PERFORMANCE (5)	10
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	35
SIGN WITH DATE	20

Result:

Thus the normalization of the given relation is created the simplified tables with suitable constraint successfully.