

Cauvery College for Women (Autonomous)

Nationally Accredited (III Cycle) with 'A' Grade by NAAC
Annamalai Nagar, Tiruchirappalli-18.



Name of the Faculty : R. SRIDEVI
Designation : Asst Professor
Department : Computer Applications
Contact Number : 9865876162
Programme : BCA
Batch : 2016-2017 Onwards
Semester : IV
Course : Database Systems
Course Code :16SCCCA4
Unit : V
Topic Covered : Normalization

Functional Dependency

Functional dependency (FD) is a set of constraints between two attributes in a relation. Functional dependency says that if two tuples have same values for attributes A_1, A_2, \dots, A_n , then those two tuples must have to have same values for attributes B_1, B_2, \dots, B_n .

Functional dependency is represented by an arrow sign (\rightarrow) that is, $X \rightarrow Y$, where X functionally determines Y . The left-hand side attributes determine the values of attributes on the right-hand side.

Armstrong's Axioms

If F is a set of functional dependencies then the closure of F , denoted as F^+ , is the set of all functional dependencies logically implied by F . Armstrong's Axioms are a set of rules, that when applied repeatedly, generates a closure of functional dependencies.

- **Reflexive rule** – If α is a set of attributes and β is subset of α , then α holds β .
- **Augmentation rule** – If $a \rightarrow b$ holds and y is attribute set, then $ay \rightarrow by$ also holds. That is adding attributes in dependencies, does not change the basic dependencies.
- **Transitivity rule** – Same as transitive rule in algebra, if $a \rightarrow b$ holds and $b \rightarrow c$ holds, then $a \rightarrow c$ also holds. $a \rightarrow b$ is called as a functionally that determines b .

Trivial Functional Dependency

- **Trivial** – If a functional dependency (FD) $X \rightarrow Y$ holds, where Y is a subset of X , then it is called a trivial FD. Trivial FDs always hold.
- **Non-trivial** – If an FD $X \rightarrow Y$ holds, where Y is not a subset of X , then it is called a non-trivial FD.
- **Completely non-trivial** – If an FD $X \rightarrow Y$ holds, where $x \cap Y = \Phi$, it is said to be a completely non-trivial FD.

Normalization

If a database design is not perfect, it may contain anomalies, which are like a bad dream for any database administrator. Managing a database with anomalies is next to impossible.

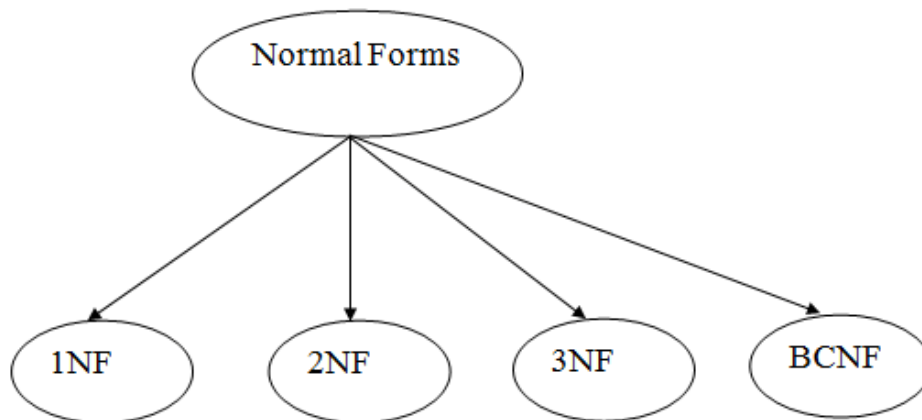
- **Update anomalies** – If data items are scattered and are not linked to each other properly, then it could lead to strange situations. For example, when we try to update one data item having its copies scattered over several places, a few instances get updated properly while a few others are left with old values. Such instances leave the database in an inconsistent state.

- **Deletion anomalies** – We tried to delete a record, but parts of it was left undeleted because of unawareness, the data is also saved somewhere else.
- **Insert anomalies** – We tried to insert data in a record that does not exist at all.

Normalization is a method to remove all these anomalies and bring the database to a consistent state.

Types of Normal Forms

There are the four types of normal forms:



First Normal Form

If a relation contain composite or multi-valued attribute, it violates first normal form or a relation is in first normal form if it does not contain any composite or multi-valued attribute. A relation is in first normal form if every attribute in that relation is atomic or **singled valued attribute**.

Example 2 –

ID	Name	Courses
1	A	c1, c2
2	E	c3
3	M	c2, c3

In the above table Course is a multi valued attribute so it is not in 1NF.

Below Table is in 1NF as there is no multi valued attribute

ID	Name	Course
1	A	c1
1	A	c2
2	E	c3
3	M	c2
3	M	c3

Second Normal Form –

To be in second normal form, a relation must be in first normal form and relation must not contain any partial dependency. A relation is in 2NF if it has **No Partial Dependency**, i.e., no non-prime attribute (attributes which are not part of any candidate key) is dependent on any proper subset of any candidate key of the table.

Partial Dependency – If the proper subset of candidate key determines non-prime attribute, it is called partial dependency.

Example 1 – Consider table-3 as following below.

STUD_NO	COURSE_NO	COURSE_FEE
1	C1	1000
2	C2	1500
1	C4	2000
4	C3	1000
4	C1	1000
2	C5	2000

{Note that, there are many courses having the same course fee. }

Here,

COURSE_FEE cannot alone decide the value of COURSE_NO or STUD_NO;

COURSE_FEE together with STUD_NO cannot decide the value of COURSE_NO;

COURSE_FEE together with COURSE_NO cannot decide the value of STUD_NO;

Hence,

COURSE_FEE would be a non-prime attribute, as it does not belong to the one only candidate key {STUD_NO, COURSE_NO} ;

But, COURSE_NO -> COURSE_FEE , i.e., COURSE_FEE is dependent on COURSE_NO,

which is a proper subset of the candidate key. Non-prime attribute COURSE_FEE is dependent on a proper subset of the candidate key, which is a partial dependency and so this relation is not in 2NF.

To convert the above relation to 2NF, we need to split the table into two tables such as :

Table 1: STUD_NO, COURSE_NO

Table 2: COURSE_NO, COURSE_FEE

Table 1		Table 2	
STUD_NO	COURSE_NO	COURSE_NO	COURSE_FEE
1	C1	C1	1000
2	C2	C2	1500
1	C4	C3	1000
4	C3	C4	2000
4	C1	C5	2000

Example 2 – Consider following functional dependencies in relation R (A, B, C, D)

AB → C [A and B together determine C]

BC → D [B and C together determine D]

In the above relation, AB is the only candidate key and there is no partial dependency, i.e., any proper subset of AB doesn't determine any non-prime attribute.

Third Normal Form –

A relation is in third normal form, if there is **no transitive dependency** for non-prime attributes as well as it is in second normal form.

A relation is in 3NF if **at least one of the following condition holds** in every non-trivial function dependency $X \rightarrow Y$

1. X is a super key.
2. Y is a prime attribute (each element of Y is part of some candidate key)

Transitive dependency – If $A \rightarrow B$ and $B \rightarrow C$ are two FDs then $A \rightarrow C$ is called transitive dependency.

- **Example** – Consider relation R(A, B, C, D, E)

A → BC,

CD → E,

B → D,

E → A

All possible candidate keys in above relation are {A, E, CD, BC} All attribute are on right sides of all functional dependencies are prime.

Boyce-Codd Normal Form (BCNF) –

A relation R is in BCNF if R is in Third Normal Form and for every FD, LHS is super key. A relation is in BCNF iff in every non-trivial functional dependency $X \rightarrow Y$, X is a super key.

- **Example 1** – Find the highest normal form of a relation R(A,B,C,D,E) with FD set as {BC→D, AC→BE, B→E}

Step 1. As we can see, $(AC)^+ = \{A, C, B, E, D\}$ but none of its subset can determine all

attribute of relation, So AC will be candidate key. A or C can't be derived from any other attribute of the relation, so there will be only 1 candidate key {AC}.

Step 2. Prime attributes are those attribute which are part of candidate key {A, C} in this example and others will be non-prime {B, D, E} in this example.

Step 3. The relation R is in 1st normal form as a relational DBMS does not allow multi-valued or composite attribute.

The relation is in 2nd normal form because $BC \rightarrow D$ is in 2nd normal form (BC is not a proper subset of candidate key AC) and $AC \rightarrow BE$ is in 2nd normal form (AC is candidate key) and $B \rightarrow E$ is in 2nd normal form (B is not a proper subset of candidate key AC).

The relation is not in 3rd normal form because in $BC \rightarrow D$ (neither BC is a super key nor D is a prime attribute) and in $B \rightarrow E$ (neither B is a super key nor E is a prime attribute) but to satisfy 3rd normal form, either LHS of an FD should be super key or RHS should be prime attribute.

So the highest normal form of relation will be 2nd Normal form.

- **Example 2** –For example consider relation R(A, B, C)

A \rightarrow BC,

B \rightarrow

A and B both are super keys so above relation is in BCNF.