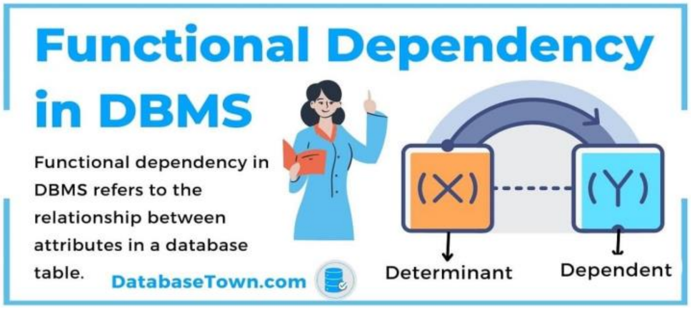
UNIT- 3

Database Design- Dependencies and Normal forms, Functional Dependencies, 1NF, 2NF, 3NF, and BCNF. Higher Normal Forms-4NF and 5NF. Transaction Management: ACID properties, Serializability, Concurrency Control, Database recovery management. Data Storage and Indexes, Hashing Techniques

1 Functional Dependencies


Functional Dependency in DBMS
Functional dependency in DBMS refers to the relationship between attributes in a database table.
DatabaseTown.com
(X)
(Y)
Determinant
Dependent

- A ***Functional Dependency*** describes a relationship **between *attributes* within a single relation.**
- An attribute is *functionally dependent* on another if we can use the value of one attribute to determine the value of another. Formally, functional dependency is defined as;
- **Functional dependency** – In a given relation R, **X and Y are attributes**. Attribute Y is functionally dependent on attribute X if each value of X determines exactly one value of Y.
- **This is represented as X→Y** $\alpha \rightarrow \beta$ ⟶ dependent

For example, consider the following relation

id	name	age
1	X	19
2	X	20
3	Y	20

Q: which of the attribute has potential to search all or any attribute .

Ans: $\boxed{id}$

fd : id → name . (True)

id → name age (True)

id → id name age

name the person where id = 1

1 → X

Tell name & age with respect to id

3 → Y and his age is 20 —

Q: is it correct?

$\boxed{\text{name} \rightarrow \text{id} ?}$ (No it is wrong)

"name can determine id" → it is wrong.

Tell me the id of X

X → 1

X → 2

Two ids are generated with one value. Hence FD is incorrect.

Full Functional Dependency - In a given relation R, X and Y are attributes. Attribute Y is fully functionally dependent on attribute X only if it is not functionally dependent on sub-set of X where X is composite in nature.

$\underbrace{\text{name age}}$ → id (Correct)

Pair of these attributes can determine id.

****Single attribute (name or age) can not determine id.**

Partial Functional Dependency –The FD (functional dependency) **A->B** happens to be a partial dependency if B is functionally dependent on A, and also B can be determined by any other proper subset of A.

id name → age

alone id → can determine age

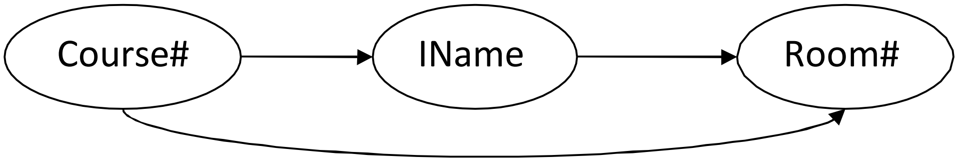
1 → 19

2 → 20

3 → 20

Transitive Dependency – In a given relation R, if attribute X determines attribute Y and attribute Y determines attribute Z, then attribute X determines attribute Z. Such a dependency is called transitive dependency.

Following example shows a transitive dependency.



Trivial Dependency -A Functional Dependency X ->Y is said to be trivial functional dependency if y is a subset of X(Y ⊆X). In other words if R.H.S of some FD is the subset of L.H.S of FD is called Trivial Functional Dependency.

Example:
AB ->A

AB ->B AB ->AB

*****Axioms:** Armstrong’s axioms are a set of rules, that when appliedrepeatedly generates a closure of functional dependencies**. Refer Class notes**

2 Attribute Closure

The set of all those attributes which can be functionally determined from an attribute set is called as closure of that attribute set.

Closure of an attribute set {X} is denoted as $\{x\}^+$.

2.1 Steps to find closure of an attribute set:

1. Add the attributes contained in the attribute set for which closure is being calculated to the result set.
2. Recursively add the attributes to the result set which can be functionally determined from the attributes already contained in the result set.

Example: Consider a relation R(A,B,C,D,E,F,G) with the functional dependencies

A−>BC BC−>DED−>F CF−>G

2.2 Closure of Attribute A:

A+ ={A}

={ABC} ∵A−>BC

={ABCDE} ∵BC−>DE

={ABCDEF} ∵D−>F

={ABCDEFG} ∵CF->G

2.3 Closure of Attribute set BC:

BC^+ ={BC}

={BCDE} ∵BC−>DE

={BCDEF} ∵D−>F

={BCDEFG} ∵CF−>G

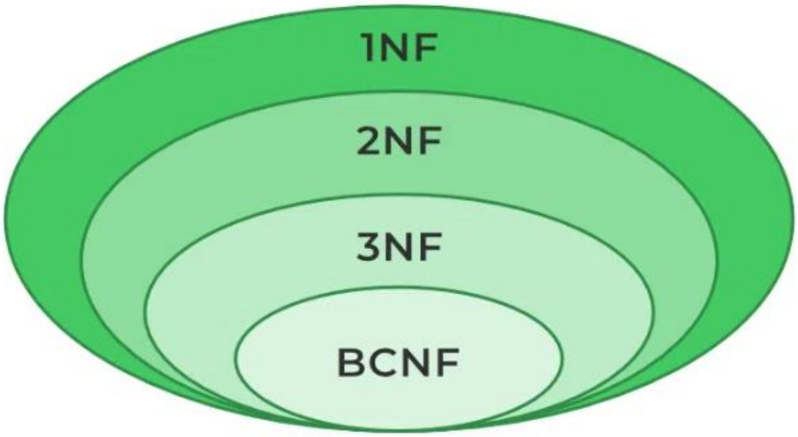
3 Normalization

Normalization is a systematic approach of decomposing tables to eliminate data redundancy and undesirable characteristics like Insertion, Update and Deletion anomalies.

- These would include two properties
  - **Lossless join property** - which guarantees that the generation of spurious tuples will not occur.
  - **Dependency preservation property** - This ensures that each functional dependency is represented in some individual relation resulting after decomposition.
- ****Prime attribute -** An attribute of relation schema R is called a prime attribute of **R if it is a member of some candidate key of R.**
- ****Non-prime attribute -** An attribute is called nonprime **if it is not a prime attribute—that is, if it is not a member of any candidate key.**

If a relation schema has more than one key, each is called a candidate key. One of the candidate keys is arbitrarily designated to be the primary key,and the others are called secondary keys.

Normal forms: The normal form is a relation refers to the highest normal form condition that it meets and hence indicates the degree to which it has been normalized. Normal forms are used to eliminate or reduce redundancy in databasetables.



BCNF in DBMS