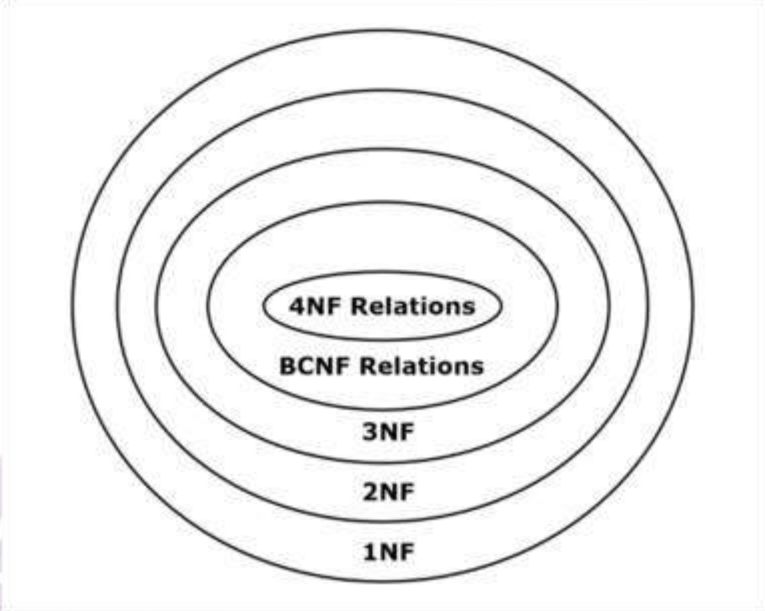
Third Normal Form in DBMS

A relation is said to be in the third normal form only if it is in the second normal form and has no transitive dependencies for non-prime attributes. Various questions are formulated in the GATE CSE question paper based on the normal form. If, for instance, a relation is not in the first normal form, then it can never be in the second normal form and, in turn, in the third normal form. Also, if a relationship is in the first normal form and not in the second normal form, it can never be in the third normal form.

The third normal form is considered the safest as it satisfies lossless join and dependency-preserving conditions for decomposing a relation. The relationship among the normal forms is depicted by the Venn diagram below:



Conditions Followed for Third Normal Form in DBMS

A relation schema R is said to be in the third normal form if it is in the second normal form and follows the below conditions for every non-trivial functional dependency X→Y in R:

- X must be the Super Key **OR**
- Y must be the prime attribute.

Also, the following dependencies of the form:

(Proper subset of candidate keys) → (Proper subset of other candidate keys)

forms redundancy and is still allowed in the third normal form.

Uses of Third Normal Form in DBMS

We employ the 3NF to minimize data duplication and achieve database integrity. The third normal form is appropriate for designing typical relational databases, which is an essential part of the GATE CSE syllabus. Most 3NF tables aren't affected by deletion, update, and insertion anomalies. A 3NF would additionally always guarantee lossless and the dependency preservation of the functional dependencies.

Moreover, among all the normal forms, the third normal form is considered the most accurate normal form because lossless join and dependency preserving is always possible. Not every relation can decompose into BCNF with dependency preserving.

How to Normalize 1NF and 2NF to 3NF?

We must ascertain whether the table contains a transitive dependency to normalize a 2NF to a 3NF. If a transitive dependency is found, the transitively dependent attributes are eliminated from the relations. This is accomplished by putting these attributes in a distinct, new relation. We also include a copy of the determinant.

Note: X → Z is referred to be a transitive dependency if X → Y and Y → Z are two functional dependencies. We eliminate these transitive dependencies while normalizing a 2NF relation to 3NF.

Practice Problem on Third Normal Form in DBMS

We have discussed the following conditions to achieve the third normal form for a relation. Now, we will see an example to understand this concept better for the GATE exam.

Example:

Consider the following functional dependencies for relation R (A, B, C, D, E, F) and find the normal form.

AB → C

C → D

CD → BE

DE → F

EF → A

Solution:

To begin with, we first find the candidate keys in DBMS for the relation. The candidate keys are: {C, AB, BEF, BDE}

Now we divide the attributes into prime and non-prime attributes.

Prime attribute set: {A, B, C, D, E, F }, all the attributes are prime. Therefore, the relation is in the third normal form or 3NF. The relation follows the condition to be in the third normal form, that is:

- X must be the Super Key **OR**
- Y must be the prime attribute.
-