

First Normal Form in DBMS

A relation is considered in first normal form (1NF) if each entry (the point where a tuple and a column intersect) contains no more than one value. The relation is in first normal form: “if and only if all underlying domains contain atomic values or single value only.”

Data redundancy rises when using the First Normal Form because each individual row will be distinct despite having many columns with the same data in multiple rows.



Rules for First Normal Form in DBMS

When creating your database, the first normal form requires you to abide by a few straightforward rules, and they are as follows:



First Rule: Single-Valued Attributes

Your table's columns should all be single-valued, meaning they shouldn't have multiple values. Let's look at the other rules for the time being, and then we'll explain this with an example.

Second Rule: The attribute domain must remain constant.

This rule is more in line with "Common Sense." The values stored in each column must be of the same kind or type.

For instance, you must not save the names of some of the people in the same column as the dates of birth of the others if the column is designated to save the dates of birth of a group of people. All records and rows should only contain the field "date of birth."

Third Rule: Give attributes and columns unique names

According to this rule, each column in a table should have a special name. This is done to prevent confusion when retrieving data or working with the stored data in any other way.

When multiple columns have the same name, the DBMS system becomes confused.

Fourth Rule: Order is irrelevant.

According to this rule, it doesn't matter what order the data are stored in your table.

Working of First Normal Form in DBMS

If a column in a table has a value that can be divided further, there shouldn't be composite values for that column; instead, a separate table with different columns and a distinct Primary Key should be created for that attribute.



For instance, a separate table called Employee address with the columns House No, City, and Pin code should exist for each employee address that can be further subdivided in the Employee table. This table should have the Employee id as its primary key.

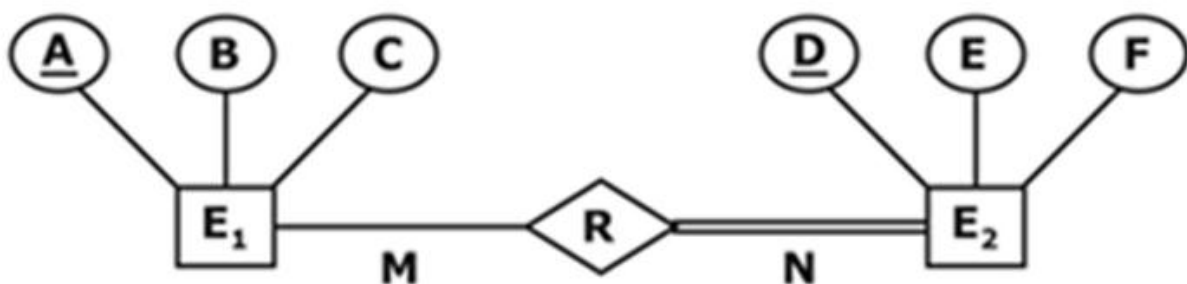
Advantages of First Normal Form in DBMS

While searching for a specific value in the database by splitting all the values of a column and then converting them again in a specific format, wasting a lot of time and memory in the process, working with tables that have columns with multiple values separated by “,” is very difficult. By making distinct rows for various values, 1NF simplifies the process.

One of 1NF’s main benefits is that it eliminates the needless repeating values by making a separate table, which means there are no problems when inserting, deleting, or updating values in the database.

Problem on First Normal Form in DBMS

Problem 1: Consider the following ER Diagram:



The number of minimum Relations which satisfy 1NF _____. (Partial participation between E1 and R should not be lost in RDBMS design).

Answer: 2

Solution:

Relation 1:

E1(A,B,C)

A is the primary key

A \rightarrow BC

Relation 2:

E2(AD,E,F)

AD is the primary key

D \rightarrow EF (by default for entity 2)

AD \rightarrow EF (from relation)

But the relation is not in 2NF, as there is a Partial Dependency(D \rightarrow EF)

so it is only in 1NF

Problem 2: Given the Relation, P (Z, Y, X, W, V, U, T) with some constraints as follows:

W \rightarrow Key attribute,

T, U, V \rightarrow Multivalued attributes

Z, Y, X \rightarrow Atomic Attributes



Then the total number of tables after decomposing the relation into 1NF is _____.

Answer: 4

Solution:

For every multivalued attribute, there will be one separate table. So, we have:

R1(W,T),

R2(W,U),

R3(W,V)

And one more table for all other attributes:

R4(W,Z,Y,X)

Thus, the total number of tables after decomposing it into 1NF is 4.

