

4th Normal Form in DBMS

Last Updated : 25 Jul, 2025

As databases grow in complexity, proper normalization becomes important to reduce data redundancy and maintain data integrity. Fourth Normal Form (4NF) is a higher level of normalization in relational database design, which deals with multivalued dependencies (MVDs).

Fourth Normal Form

Course	Instructor	TextBook_Author
Management	X	Churchill
Management	Y	Peters
Management	Z	Peters
Finance	A	Weston
Finance	A	Gilbert

Course \longrightarrow **Instructor**

Course \longrightarrow **TextBook_Author**

Multivalued Dependency

Multivalued Dependency

A multivalued dependency occurs in a relation when one attribute determines multiple independent values of another attribute, independent of other attributes. A multivalued dependency always requires at least three attributes because it consists of at least two attributes that are dependent on a third.

For a dependency $A \twoheadrightarrow B$, if for a single value of A, multiple values of B exist, then the table may have a multi-valued dependency. The table should have at least 3 attributes and B and C should be independent for $A \twoheadrightarrow B$ multivalued dependency.

Example: A course can have multiple instructors, a course can also have multiple textbook authors but instructors and authors are independent of each other. This creates two independent multivalued dependencies:

Course \twoheadrightarrow Instructor

Course \twoheadrightarrow TextBook_Author

If stored in the same table, this creates redundant combinations and data anomalies. A multivalued dependency is a generalization of a functional dependency, but they are not the same.

Fourth Normal Form (4NF)

The Fourth Normal Form (4NF) is a level of database normalization where there are no non-trivial multivalued dependencies other than a candidate key. It builds on the first three normal forms (1NF, 2NF and 3NF) and the [Boyce-Codd Normal Form \(BCNF\)](#). It states that, in addition to a database meeting the requirements of BCNF, it must not contain more than one multivalued dependency. It is an extension of Boyce-Codd Normal Form (BCNF) and ensures that a relation does not contain multiple independent one-to-many relationships within a single table.

Fourth Normal Form

Course	Instructor	TextBook_Author
Management	X	Churchil
Management	Y	Peters
Management	Z	Peters
Finance	A	Weston
Finance	A	Gilbert

Course

TextBook Author

Course	Instructor
Management	X
Management	Y
Management	Z
Finance	A

Course	Instructor
Management	Churchil
Management	Peters
Finance	Weston
Finance	Gilbert

4th Normal Form

Properties

A relation R is in 4NF if and only if the following conditions are satisfied:

1. It should be in the Boyce-Codd Normal Form (BCNF).
2. The table should not have any Multi-valued Dependency.

Key Idea: 4NF eliminates redundancy caused by multivalued dependencies by separating independent one-to-many relationships into different tables.

A table with a multivalued dependency violates the normalization standard of the Fourth Normal Form (4NF) because it creates unnecessary redundancies and can contribute to inconsistent data. To bring this up to 4NF, it is necessary to break this information into two tables.

Example: Consider the database table of a class that has two relations R1 contains student ID(SID) and student name (SNAME) and R2 contains course id(CID) and course name (CNAME).

Table R:

Course	Instructor	TextBook_Author
Management	X	Churchill
Management	Y	Peters
Management	Z	Peters
Finance	A	Weston
Finance	A	Gilbert

Problem:

- Each Course has multiple Instructors.
- Each Course has multiple TextBook_Author.
- But Instructor and TextBook_Author are not related to each other.
- This causes repetition of combinations, violating 4NF.

Solution: To remove the MVDs and bring the relation to Fourth Normal Form, we split the original table into two separate tables, each handling one multivalued dependency. This improves data integrity and removes redundancy.

Table R1:

Course	Instructor
Management	X
Management	Y
Management	Z

Course	Instructor
Finance	A

This table shows which instructor teaches which course.

Table R2:

Course	TextBook_Author
Management	Churchill
Management	Peters
Finance	Weston
Finance	Gilbert

Result: *Now, the 4NF is Achieved*

Benefits of Decomposition:

1. No repetition of unrelated attribute combinations.

- In a non 4NF table, if two attributes are independently related to a third, their combinations get repeated unnecessarily.
- This leads to a cartesian product effect, lots of rows just to represent all combinations.

Example: If a course has 3 instructors and 2 textbook authors, we get $3 \times 2 = 6$ rows, even though there's no link between instructors and authors.

After 4NF decomposition: Instructors and authors are stored in separate tables, so:

- Instructors: 3 rows, Authors: 2 rows.No redundant pairings between them.
- Each table contains data with a single multivalued dependency.
- Both tables are now in BCNF and 4NF.
- Ensures cleaner design, efficient storage and no anomalies.

2. Each Table Contains Data with a Single Multivalued Dependency

- Every decomposed table focuses on only one multivalued relationship.
- There is one clear dependency per table (e.g., Course ->-> Instructor OR Course ->-> Textbook_Author), not both.

Why it's important:

- It simplifies understanding, querying and maintaining the data.
- Each relation represents one fact, reducing logical complexity.
- This aligns with principle of separation of concerns - one table, one purpose.

3. Both Tables Are Now in BCNF and 4NF

After decomposition:

- There are no partial, transitive or multivalued dependencies.
- All attributes are functionally dependent only on the whole key.

Result: The structure now meets

- Boyce-Codd Normal Form (BCNF) as Every determinant is a candidate key.
- Fourth Normal Form (4NF) as No non-trivial MVDs exist.
- Tables are well-structured, normalized and reliable.

4. Ensures Cleaner Design, Efficient Storage and No Anomalies

- Each table is focused and easier to read.
- Developers and DBAs can understand the schema without confusion.

Efficient Storage:

- Redundant rows are eliminated.
- Fewer rows corresponds to Less storage space and so Faster performance.

No Anomalies:

- Insertion anomaly: You can insert a new instructor without needing a textbook.
- Deletion anomaly: Deleting a textbook doesn't remove the instructor.
- Update anomaly: Update happens in one place only and no risk of mismatched data.

Note: *Decomposing tables to eliminate multivalued dependencies isn't just about "following rules" , but it's about making your data model more logical, efficient and future-proof.*