Normalization

Normalization in database management systems (DBMS) is the process of organizing data in a database to reduce redundancy, improve data integrity and avoid upadation inseration and deletion anomalies. There are different normal forms, each building on the previous one.

Types:

1. **First Normal Form (1NF)**:
    - In 1NF, each column in a table contains atomic (indivisible) values, and there are no repeating groups.
    - Example:

OrderID	Customer	Item 1	Item 2	Item 3
1001	John	Apple	Orange	Banana
1002	Mary	Pear	Apple	

    The above table is not in 1NF because the "Item" columns contain repeating groups. To normalize it to 1NF, we can create a separate table for items:

    Orders Table

OrderID	Customer
1001	John
1002	Mary

    Items Table

OrderID	Item
1001	Apple
1001	Orange
1001	Banana
1002	Pear
1002	Apple

2. **Second Normal Form (2NF)**:
    - In 2NF, the table is in 1NF, and all non-key columns are fully functionally dependent on the entire primary key.
    - Example:

    Orders Table

OrderID	Customer	Employee	EmployeeLocation
1001	John	Smith	New York
1002	Mary	Johnson	Los Angeles

In this example, the table is not in 2NF because "EmployeeLocation" is dependent on "Employee," which is only part of the primary key. To normalize it to 2NF, we can create a separate table for employee information:

Orders Table

OrderID	Customer	EmployeeID
1001	John	1
1002	Mary	2

Employees Table

EmployeeID	Employee	EmployeeLocation
1	Smith	New York
2	Johnson	Los Angeles

3. **Third Normal Form (3NF)**:
    - In 3NF, the table is in 2NF, and there are no transitive dependencies between non-key columns.
    - Example:

      Orders Table

OrderID	Customer	Product	Category	CategoryDescription
1001	John	Apple	Fruits	Fruits are healthy.
1002	Mary	Pear	Fruits	Fruits are healthy.
1003	Alex	Television	Electronics	Electronic devices.

In this example, the table is not in 3NF because "CategoryDescription" is dependent on "Category," which is not part of the primary key. To normalize it to 3NF, we can create a separate table for categories:

Orders Table

OrderID	Customer	Product	CategoryID
1001	John	Apple	1
1002	Mary	Pear	1
1003	Alex	Television	2

Categories Table

CategoryID	Category	CategoryDescription
1	Fruits	Fruits are healthy.
2	Electronics	Electronic devices.

4. **Boyce-Codd Normal Form (BCNF)**:

   - BCNF is an extension of 3NF and deals with the issue of functional dependencies involving candidate keys.
   - In BCNF, every non-key attribute is fully functionally dependent on the candidate key, and there are no partial dependencies.
   - A table is in BCNF if, for every non-trivial functional dependency (X -> Y), X is a superkey.

   Example:

   Consider the following table:

   Students Table

StudentID	Course	ProfessorID	ProfessorName
1	Math	101	Smith
2	Physics	102	Johnson
3	Math	101	Smith
4	History	103	Brown

   In this example, the table is not in BCNF because "ProfessorName" is functionally dependent on "ProfessorID," which is not a candidate key. To bring it to BCNF, we split the table into two:

   Students Table

StudentID	Course	ProfessorID
1	Math	101
2	Physics	102
3	Math	101
4	History	103

   Professors Table

ProfessorID	ProfessorName
101	Smith
102	Johnson
103	Brown

   Now, both tables are in BCNF, and there are no partial dependencies.

5. **Fourth Normal Form (4NF)**:

   - 4NF is an extension of BCNF and deals with multivalued dependencies.
   - A table is in 4NF if it is in BCNF, and there are no non-trivial multivalued dependencies.

Example:

Consider the following table:

Authors Table

BookID	Author	Genres
1	John	Fiction, Comedy
2	Mary	Mystery
3	John	Drama, Romance

In this example, the table is not in 4NF because there are multivalued dependencies in the "Genres" column. To bring it to 4NF, we split the table into two:

Authors Table

BookID	Author
1	John
2	Mary
3	John

Genres Table

BookID	Genre
1	Fiction
1	Comedy
2	Mystery
3	Drama
3	Romance

Now, both tables are in 4NF, and there are no multivalued dependencies.