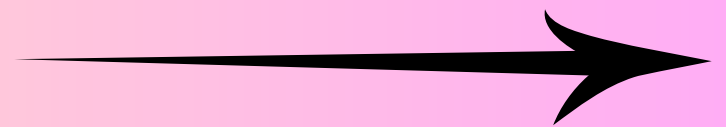


Understanding Normalization in SQL

@Vaishnavi Dauale





What is Normalization?

- Normalization is the process of structuring data in a database to eliminate redundancy and avoid anomalies like Insertion, Update, and Deletion anomalies.
- It involves dividing large tables into smaller, related tables and establishing clear relationships between them.



Why Use Normalization?

- By reducing redundancy and ensuring data dependencies are logical and consistent, normalization enhances data integrity.
- It simplifies data maintenance tasks such as updates, deletions, and insertions, thereby improving overall database management efficiency.
- Additionally, normalization optimizes query performance by streamlining data retrieval



Types of Normal Forms

1NF : Atomicity by requiring each column to contain unique, indivisible values.

Students

StudentID	Name	Course
1	Alice	Math
1	Alice	Science
2	Bob	History
2	Bob	Art



2NF :

- Remove partial dependencies where non-key attributes depend on only part of a composite key.

Orders

OrderID	CustomerID
1	C01
2	C02

OrderDetails

OrderID	ProductID
1	101
2	102

Customers

CustomerID	CustomerName
C01	Alice
C02	Bob



3NF :

- Eliminate transitive dependencies where non-key attributes depend on other non-key attributes.

Employees

EmployeeID	Name	DepartmentID
1	Alice	D01
2	Bob	D02

Departments

DepartmentID	DepartmentName
D01	HR
D02	Finance



4NF :

- Eliminate multi-valued dependencies by ensuring that records do not contain two or more independent multi-valued facts.

ProjectAssignments

ProjectID	EmployeeID
1	E01

EmployeeSkills

EmployeeID	Skill
E01	Java
E01	Python





5NF :

5nf - Decompose tables to remove redundancy by ensuring every join dependency is implied by candidate keys.

DKNF :

Eliminate all constraints except domain and key constraints.





Advantages

- **Eliminate all constraints except domain and key constraints**Minimizes duplicate data across the database, leading to more efficient storage and streamlined data management.
- **Organizes data into atomic (indivisible) values, ensuring that each column in a table holds only unique, non-repeating information.**



- **implifies data maintenance tasks such as updates, deletions, and insertions by reducing data duplication and ensuring that data is logically organized**
- **Optimizes the database structure, leading to potentially faster and more efficient data retrieval due to reduced data volume and better-structured queries..**



10

Ensures that complex relationships are handled correctly, with data dependencies clearly defined.



Disadvantages

- Normalization can lead to a more complex database schema with a larger number of smaller, related tables, which can be harder to manage and understand.
- The design and maintenance of the database become more challenging due to the need for a detailed understanding of the data relationships and dependencies.

11

Thank You

@Vaishnavi Dauale

