

Normalization

- Normalization is the process of removing duplicate data from a table.
- It divides a larger table into smaller tables and links them using relationships.

Why Do We Need Normalization?

- The main reason for normalization is removing anomalies.

What is an Anomaly?

- An anomaly is a side effect caused by duplicate data or large tables.

Example Table

S_id	s_name	s_contact	c_id	c_name	f_name	f_contact
101	John	1234567890	201	Math	Smith	9876543210
102	Alice	1234567891	202	Science	Brown	9876543211
101	John	1234567890	203	History	Smith	9876543210

Types of Anomalies

- Insertion Anomaly:
Occurs when you cannot insert a new record because some required data is missing.
For example, if s_id is the primary key and cannot be null, we cannot insert a new course without valid student data.
- Update Anomaly:
Occurs when updating a single piece of data requires multiple changes.
For example, if faculty data is repeated several times, updating the faculty contact requires multiple updates.
- Deletion Anomaly:
Occurs when deleting data unintentionally removes other important data.
For example, deleting a student entry might also delete related course and faculty information.

To avoid these anomalies, we divide one big table into smaller related tables.

Smaller Normalized Tables Example

S_id	s_name	s_contact
101	John	1234567890
102	Alice	1234567891

C_id	c_name
201	Math
202	Science
203	History

f_name	f_contact
Smith	9876543210
Brown	9876543211

Types of Functional Dependencies

1. Total Functional Dependency:
All non-key attributes depend on the entire primary key attribute.
Example: $R=\{empno,ename,job,sal,hiredate\}$ $R=\{empno,ename,job,sal,hiredate\}$
2. Partial Functional Dependency:
Non-key attributes depend on part of a composite primary key.
Example: $R=\{empno,empname,job,sal,deptno,dname,loc\}$ $R=\{empno,empname,job,sal,deptno,dname,loc\}$
3. Transitive Functional Dependency:
A non-key attribute depends on another non-key attribute, which depends on the key.
Example: $R=\{empno,ename,job,sal,deptpincode\}$ $R=\{empno,ename,job,sal,deptpincode\}$

Normalization Forms

First Normal Form (1NF)

- A table is in 1NF if there is no duplicate data.

- It has a primary key.
- All values are atomic (indivisible).

Example:

Roll_no	Name	Course
1	James	Laptop
2	Rodes	Tablet
3	Williams	Shirts
4	Smith	Watches

Second Normal Form (2NF)

- Table is in 1NF.
- No partial dependency of any column on part of the primary key.

Example:

Student Table

student_id	student_name
121	James
122	Rodes

Course Table

course_id	course_name
32	Laptop
33	Tablet

Third Normal Form (3NF)

- Table is in 2NF.
- There is no transitive dependency between non-key attributes.

Example:

emp_no	emp_name	job	salary	dept_pincode
101	Alice	Dev	50000	12345

Summary

- Once tables are normalized, relations can be built using foreign keys.
- Normalization helps reduce redundancy and anomalies in the database.