Database Normalization

What is Database Normalization?

Database normalization is the technique to remove the redundant or duplicate data and improve data integrity.

What is redundant data?

There can be two types of redundant or duplicate data in tables:

1. **Row-level**

```
| Student_id | Student_name | Age | |------------|--------------|----
-| | 1 | Yash | 27 | | 2 | Linzy | 27 | | 1 | Yash | 27 |
```

Row 1 and Row 3 are duplicate rows and this is row-level duplicate data. To solve, we set Student_ID as Primary key.

```
| Student_id | Student_name | Age | |------------|--------------|----
-| | 1 | Yash | 27 | | 2 | Linzy | 27 | Primary-key -> Student_id
```

(Primary key is the unique and NOT NULL value)

2. **Column-level**

```
| Student_id | Student_name | Course_id | Course_name | Faculty_id |
Faculty_name | Salary | |------------|--------------|-----------|----
---------|------------|--------------|--------| | 1 | Yash | C1 |
DBMS | F1 | John | 1000 | | 2 | Eric | C2 | DSA | F2 | XYS | 2000 | |
3 | Linzy | C1 | DBMS | F1 | John | 1000 | | 4 | Rob | C1 | DBMS | F1
| John | 1000 |
```

In this table, there are no row-level duplicate data but there is column-level duplicate data where Course_id, Course_name, Faculty_id, Faculty_name have same values.

Why normalization is required?

Normalization helps in removing the redundant data and without normalization we may face issues such as:

1. Insertion anomaly
2. Update Anomaly
3. Delete Anomaly

Let’s take above table as example-

```
| Student_id | Student_name | Course_id | Course_name | Faculty_id |
Faculty_name | Salary | |------------|--------------|-----------|----
---------|------------|--------------|--------| | 1 | Yash | C1 |
DBMS | F1 | John | 1000 | | 2 | Eric | C2 | DSA | F2 | XYS | 2000 | |
3 | Linzy | C1 | DBMS | F1 | John | 1000 | | 4 | Rob | C1 | DBMS | F1
| John | 1000 |
```

Insertion Anomaly

Adding a new course “Intro to SQL” into the above table but since there is no students enrolled yet, we can not insert it. This is insertion anomaly.

Update Anomaly

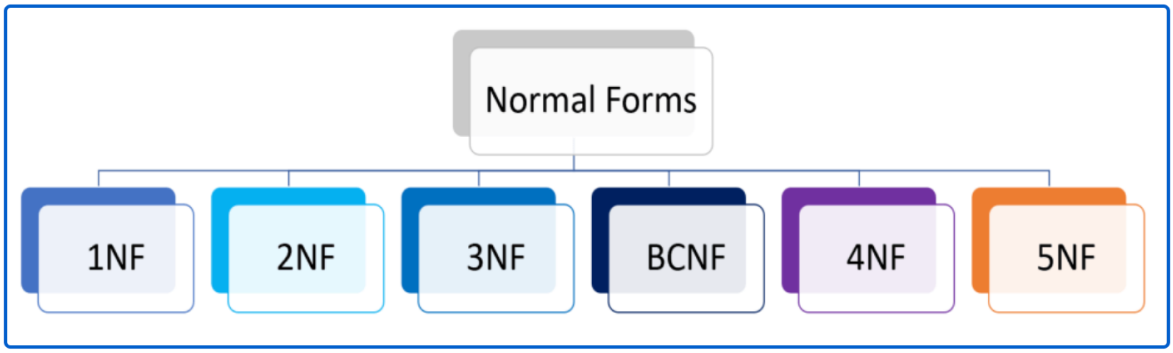
Let’s suppose we want to update the “salary” of faculty “F1” from 1000 to 1500. Updating a value is easy in this case but we have to update all the rows with faculty “F1” and this is a costly operation when working with million rows of data.

Deletion Anomaly

Let’s suppose, we want to delete the record of Student “Eric” from the table. It will not throw an error but what if “Eric” was the only student in the “DSA” course. Now, we can not retrieve the data for course “DSA”.

To solve all the above anomalies, we need to normalize the Database.

Normalization has various forms using which we can normalize the data and you can find those normal forms in the next post. Thank you for reading. 🙂


Normal Forms
1NF
2NF
3NF
BCNF
4NF
5NF