**NORMALISATION OF DATABASE**

* NORMALISATION: - It is a Database Design Technique that organises tables in a manner that reduces redundancy(useless) and dependency of data.
* To avoid insertion, update, deletion anomaly.
* (ANOMALY= something that is not according to standard or normal)
* It divides larger tables into smaller tables and links them using relationship (Primary key and Foreign key)

In short :-

—If our table is not in std format or not contain unique values in all columns then we cannot easily insert, delete, or update any single row or column that’s why we need to do NORMALISATION….

Most Commonly used normal forms:-

* FIRST NORMAL FORM(1NF)
* SECOND NORMAL FORM(2NF)—Must fulfil the criteria of 1NF
* THIRD NORMAL FORM(3NF) -- Must fulfil the criteria of 1NF AND 2NF
* BOYCE AND CODD NORMAL FORM(BCNF) -- Must fulfil the criteria of 1NF AND 2NF AND 3NF
* FOURTH NORMAL FORM(4NF)
* FIFTH NORMAL FORM(5NF)
* SIXTH NORMAL FORM(6NF)

IN DETAILS:-

**1NF**

* Each attribute should contain one value
* Each attribute(column) contains atomic values which cannot be split further

**2NF**

* It should be in 1NF
* No partial dependency (a=b)

**3NF**

* It should be in 1NF and 2NF
* Remove Transitive dependency (a=b, b=c, i.e. c=a)

**BCNF**

* Also referred as 3.5NF
* It is advance version of 3.5NF
* It should be in 3NF
* In every functional dependency x should depend on y, and x should be ‘SUPER KEY’ of a table
* SUPER KEY:-

If x determines y in one table, then x is super key of that table and y determines z in another table, then y is super key of that table

**4NF**

* It should be in BCNF
* Remove Multi-Valued Dependency
* If A derives B, for a single value of A, mora than one value of B exists
* Table should have at-least 3 columns
* For this table with A, B, C columns, B and C should be independent

**5NF**

* It should be in 4NF
* It should not have JOIN Dependency, If join dependency exists then decomposed the table
* It also known as PJNF (Project Join Normal Form)
* One to one relation with each column (Ex- BINARY RELATIONSHIP-

Supplier-Customer

Customer- Product

Product- Supplier

* JOIN Dependency= If a table decomposed into smaller tables that leads to loss of information getting created then we should not decomposed the table because that will lead to incorrect information, then we stick to the original table as per the 5NF, BUT if breaking down the table doesn’t lead to info. Loss WE can still verify all facts about data then we should decompose the table

**6NF**

* It is the most advanced level of Database Normalisation
* A table in a 6NF if-
* It is already in 5NF
* It cannot be decomposed further without losing info. And all JOIN dependency are trivial (a little value or importance)
* IN SHORT- In simple terms, 6NF is used when the data needs to be split into the smallest possible tables often for managing time-variant data or historical data records ensuring each piece of data stands independently and efficiently tracks changes over time