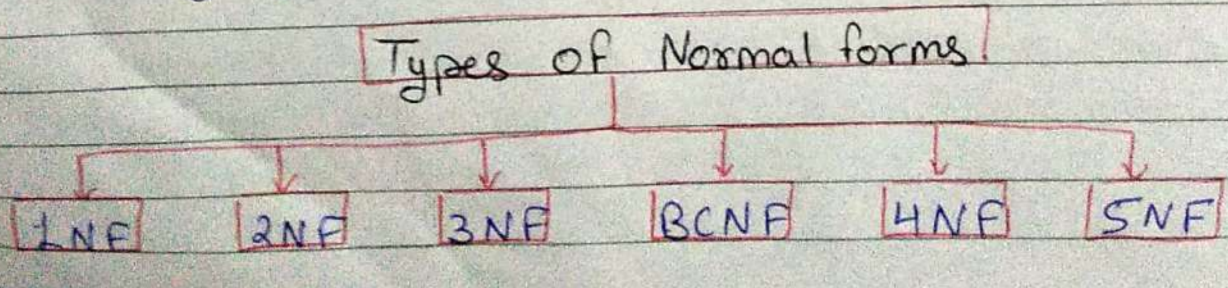
Normalization

- Normalization is the process of organizing the data in the database.
- Normalization is used to minimize the redundancy from a relation or set of relations
- It is used to eliminate undesirable characteristics like insertion, Update and Deletion Anomalies.
- Normalization divides the Larger table into Smaller and links them using relationships.
- The normal form is used to reduce redundancy from the database table.

When we normalize the database, we have four goals:

1) Arranging data into Logical groupings such that Each group describes a small part of the whole.
2) minimizing the amount of duplicate data, Called redundancy, stored in a database.
3) organizing the data such that, when you modify it, you make the changes only in one place.
4) Building a database in which you can access and manipulate the data quickly and efficiently without Compromissing the integrity of the data in storage.



1NF (first Normal form) ⇒ • A relation is in 1NF if it Contains an atomic values.

- It Eliminate Repeating Groups.

2NF (second Normal form) ⇒ • A relation will be in 2NF if it is in 1NF and all non-key attributes are fully functional dependent on the primary key.

- It Eliminate Partial functional dependency.

3NF (Third Normal form) ⇒ • A Relation will be in 3NF if it is in 2NF and no transitive dependency Exists.

- It Eliminate Transitive dependency.

BCNF (Boyce Codd's normal form) ⇒ • A Stronger Definition of 3NF is known as Boyce Codd's normal form.

- It is also Called 3.5 NF.

4NF (fourth Normal form) ⇒ • A relation will be in 4NF if it is in Boyce Codd's Normal form and has no multi-valued dependency

- Eliminate multi-Values Dependency.

5NF (fifth Normal form) ⇒ • A relation is in 5NF if It is in 4NF and does not Contain any

- join dependency, joining should be Lossless.
- Eliminate Join Dependency.

Advantages & Disadvantages of Normalization

Advantages of Normalization.

- Normalization helps to minimize data redundancy.
- Greater overall database organization.
- Data Consistency within the database.
- much more flexible database design.
- fewer indexes per table ensure faster maintenance tasks(index rebuilds).
- Relizes the option of Joining only the tables that are needed.
- A better handle on database Security.
- Increase Storage efficiency.
- Speed up data access.

Disadvantages of Normalization.

- You Cannot Start building the database before knowing what the user needs.
- Requires Much CPU, memory and I/O process thus normalized data gives reduced database performance
- Requires more joins to get the desired result. A poorly-written query can bring the database down.
- It is very time-Consuming and Difficult to normalize relations of a higher degree.
- Careless decomposition may lead to a bad database design, Leading to Serious problems.

First Normal Form (1NF)

- A relation will be 1NF if it Contains an atomic value.
- It States that an attribute of a table Cannot hold multiple values. It must hold only Single-valued attribute.
- First Normal form disallow the multi-valued attribute, Composite attribute and their Combinations.

Ex ⇒ Relation Student is not in 1NF because of multi-Valued attribute ~~Std~~-Phone.

Student table:

Stud_id	Stud_name	Stud-Phone	Stud_Branch
11	Kamal	7276854823, 3214172829	IT
12	karan	8762547890	CS
13	Ravi	32456278, 34563578	ES
14	Ram	8176345630	ME
15	Komal	3456873200	IT

The Decomposition of the Student table into 1NF as:

Stud_id	Stud_name	Stud-Phone	Stud-Branch
11	Kamal	7276854823	IT
11	Kamal	3214172829	IT
12	Karan	8762547890	CS
13	Ravi	32456278	ES
13	Ravi	34563578	ES
14	Ram	8176345630	ME
15	Komal	3456873200	IT

Second Normal form (2NF)

- In the 2NF, relation must be in 1NF.
- No attributes of the table should be functionally dependent on only one part of a Concatenated primary key.
- In the second normal form, all non-key attributes are fully functional dependent on the primary key.

Ex ⇒ 2NF is based on the concept of full functional Dependency $\boxed{X \longrightarrow Y}$ is a fully function Dependency (FFD) if removal of any attribute (A) from X means that the dependency does not hold any more.

In the Book-order table Such as

order No.	title	Qty	Unit Price
1	Computer Network	1	250
1	Java	1	275
1	DBMS	2	295
2	multimedia	1	300
2	Data Structure	1	190
3	DBMS	1	295
3	multimedia	2	300
3	Computer Network	5	250

It is not in 2NF because it hold Partial function Dependency and fully function dependency.

title ⟶ Unit Price

P.F.D.

and

OrderNo, Title → Qty } F.F.D.

$$\text{OrderNo} \xrightarrow{\times} \text{Qty}$$

$$\text{Title} \xrightarrow{\times} \text{Qty}$$

Now we will convert the given table in 2NF to decompose the given table into two sub table such as:

Order-master table

Order No.	Title	Qty
1	Computer Network	1
1	Java	1
1	DBMS	2
2	Multimedia	1
2	Data Structure	1
3	multimedia	2
3	Computer Network	5

Book_Master table

Title	unit Price
Computer Network	250
Java	275
DBMS	295
Multimedia	300
Data Structure	190

Third Normal form (3NF)

- A relation is in 3NF, if it is in 2NF and no non-prime attribute functionally dependent on other non-prime attributes.
- A relation is in 3NF, if it is in 2NF and no attributes of the table should be transitively functionally dependent on the primary key.
- 3NF is used to reduce the data duplication. It is also used to achieve the data integrity.
- If there is no transitive dependency for non-prime attributes, then the relation must be in third normal form.

A relation is in third normal form if it holds atleast one of the following conditions for Every non-trivial function dependency X → Y.

1→ X is Super key

2→ Y is a prime attribute suchas Each Element of Y is part of some Candidate key.

Ex ⇒ Employee_Department_Location table

EmpNo	E_Name	Sal	Depart_Name	Depart_Location
1001	Vishal	7500	Accounts	102
1002	Amit	5000	Sales	104
1003	Anuj	10000	Accounts	102
1004	Vikas	4500	Sales	104
1005	Sumit	6500	Store	106

Table Not in 3NF because it hold the transitive dependency.

EmpNo → Dept-Name → Dept-Location

EmpNO → Dept-Location

To make it in 3NF we decompose and remove the transitive dependency. so we convert the given table in 3NF decompose two sub table Such as :-

(Table1) Employee - Depatment

EmpNO	E-Name	Sal	Dep_Name
1001	Vishal	7500	ACCount
1002	Amit	5000	Sales
1003	Anuj	10000	ACCounts
1004	Vikas	4500	Sales
1005	Sumit	6500	Store

Table 2 Department- Location

Dept-Name	Dept- Location
ACCounts	102
Sales	104
Store	106

Table converted in 3NF.

Boyce Codd Normal form (BCNF)

- BCNF is the advance version of 3NF. It is stricter than 3NF.
- A table is in BCNF if every functional dependency X → Y, X is the super key of the table.
- For BCNF, the table should be in 3NF, and for every FD, LHS is super key.

Example :- Let's assume there is a company where employees work in more than one department.

Employee table

Emp-id	Emp-Country	Emp-Dept	Dept-type	Emp-Dept-No
264	India	Designing	D394	283
264	India	Testing	D394	300
364	UK	Stores	D283	232
364	UK	Developing	D283	549

In the above table Functional dependencies are as follows :-

Emp-id → Emp-Country

Emp-Dept → { Dept-type, Emp-Dept-No }

Candidate key : { ~~Emp-type,~~ Emp-id, Emp-Dept }

the table is not in BCNF because neither Emp-dept nor Emp-id alone are keys.

To convert the given table into BCNF, we decompose it into three tables.

Emp_Country table:

Emp_id	Emp_Country
264	India
364	UK

Emp-Dept table:

Emp-Dept	Dept_type	Emp-Dept-No
Designing	D394	283
Testing	D394	300
Stores	D283	232
Developing	D283	549

Emp-Dept_mapping table:

Emp-id	Emp_Dept
D394	283
D394	300
D283	232
D28.3	549

functional dependencies:

Emp_id → Emp_Country

Emp-Dept → {Dept_type, Emp_Dept_No}

Candidate keys:

for the first table: Emp_id

for the Second table: Emp-Dept

for the third table: {Emp_id, Emp_Dept}

Now, this is in BCNF because left side part of both the functional dependencies is a key.

Fourth Normal form (4NF)

- A relation will be in 4NF if it is in Boyce codd normal form (BCNF) and has no multi-valued dependency.
- MVD (multi-value dependency) occurs when two or more independent multi-valued facts about the same attribute occurs within the same relation.
- MVD is denoted by

$$X \rightarrow \rightarrow Y$$

It will be readers "X there is a multi-valued dependency of Y" or "X multi-determines Y".

Example ⇒ Faculty

faculty	Subject	Committee
Kailash	DBMS	Placement
Kailash	Java	Placement
Kailash	C	Placement
Kailash	DBMS	Scholarship
Kailash	Java	Scholarship
Kailash	C	Scholarship

The given faculty table is in 3NF, but the Subject and Committee are two independent Entity. Hence there is no relationship between Subject and Committee.

In the faculty relation, a faculty with faculty name Kailash Contains three Subject DBMS, Java and C, and two Committee placement and scholarship. so there is a multi-valued dependency on faculty name, which leads to unnecessary repetition of data.

Kailash → → Placement
Kailash → → Scholarship.

So to make the above table into 4NF, we can decompose it into two tables:

Table 1 faculty_Course

faculty	Subject
Kailash	DBMS
Kailash	Java
Kailash	C

Table 2 faculty - Committee

faculty	Committee
Kailash	Placement
Kailash	Scholarship

co

Fifth Normal form (5NF)

- The 5NF (fifth Normal form) is also known as project-join Normal form.
- A relations is in fifth Normal form (5NF), if it is in 4NF, and won't have Lossless decomposition into Smaller tables.
- 5NF is Satisfied when all the tables are broken into a many tables as possible in order to avoid redundancy After that you Combined these all tables if it is Equal to original table then 5NF.
- You Can also Consider that a relation is in 5NF, if the Candidate key implies Every join dependency in it.

Example ⇒ faculty (original table)

faculty	Subject	Committee
Kailash	DBMS	Placement
Kailash	Java	placement
Kailash	C	Placement
Kailash	DBMS	Scholarship
Kailash	Java	Scholarship
Kailash	C	Scholarship

the given table is not in 4Nf and 5NF first we Convert it in 4NF with Converting it two Sub table.

Table1 faculty - Subject

faculty	Subject.
Kailash	DBMS
kailash	Java
Kailash	C

Table 2 faculty - Committee

faculty	Committee
Kailash	Placement
Kailash	Scholarship

To Convert it in 5NF, we join both table1 and table2 if it give the result Same as original table (faculty) then it's in 5NF otherwise not in 5 NF.

Table1 + Table2

faculty	Subject	Committee
Kailash	DBMS	Placement
Kailash	DBMS	Scholarship
Kailash	Java	Placement
Kailash	Java	Scholarship
Kailash	C	Placement
Kailash	C	Scholarship

is Equal to original tab

So it is in 5NF