NORMALISATION

TYPES OF ATTRIBUTES :—

- ① Key Attribute
- ② Non key attribute
- ③ Primary key attribute
- ④ Non Prime key attribute
- ⑤ Super key attribute
- ⑥ Composite key attribute

1) KEY ATTRIBUTE

- Also known as "Candidate key".
- It is the unique attribute present in the table.
- example :—

StudID, Sname, Scon, Sloc, Sbranch, Yop, Semail.

2) NON KEY ATTRIBUTE

- Non key attributes are not unique.
- example :—

Sname, Sloc, Sbranch, Yop.

other than key attribute all the remaining.

3) PRIME PRIMARY KEY ATTRIBUTE

- key's which is eligible for becoming a primary key but among them only one can be selected as Prime key attribute.
- example :—

StudID, Scon, Semail.

4) NON-PRIME KEY ATTRIBUTE:-

- All the remaining ~~Primary~~ key's which satisfy the characteristics of primary key other than prime key attribute.
- example:-
  Scon , Semail.

5) SUPER KEY ATTRIBUTE:-

- Combination of two or more key Attributes
- example:-
  Scon , Semail , Stud ID + Semail + Scon
  (Addition of key attributes)

6) COMPOSITE KEY ATTRIBUTE:-

- Combination of two or more non-key attribute.
- example:-
  Sname , Sloc , Sbranch , Sname + Sloc + Sbranch

7) FOREIGN KEY:-

- It is used to have connection between two tables i.e. from one table to another table.

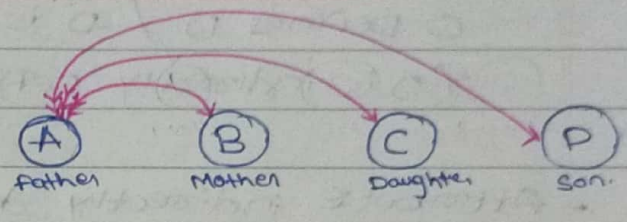
TYPES OF DEPENDANCIES :-

1. Total functional dependancies.
2. Partial functional dependancies.
3. Transitive functional dependancies.

(1) TOTAL FUNCTIONAL DEPENDANCIES :-

$A \longrightarrow B$
$A \longrightarrow C$
$A \longrightarrow D$

ex.



i.e.

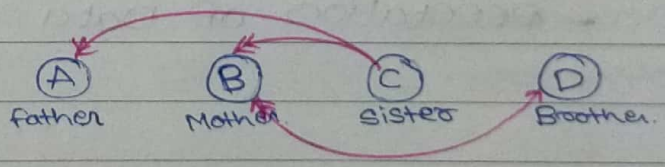
A defines B / B is dependent on A.
A defines C / C is dependent on A.
A defines D / D is dependent on A.

- All the attributes depends on one attribute.

(2) PARTIAL FUNCTIONAL DEPENDANCIES :-

$A \longrightarrow C$
$B \longrightarrow D$
$B \longrightarrow C$

ex.



i.e.

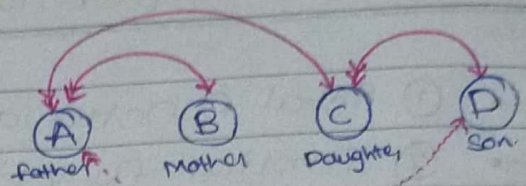
A defines C / C is dependent on A.
B defines D / D is dependent on B.
B defines C / C is dependent on B.

- It means composite key attribute depends on two attributes.

3) TRANSITIVE FUNCTIONAL DEPENDANCIES:-

A → B
A → C
C → D

ex.



i.e.

A defines B / B is dependent on A.
A defines C / C is dependent on A.
C defines D / D is dependent on C
(D is indirectly defined by A)

- Attribute indirectly depends on another Attribute.

- one non key attribute depends on another non key attribute.

• REDUNDANCIES / REDUNDANCY

- Repetation of Data or Duplication of Data.

• ANOMALIES :-

- They can be termed as side effects that we can have in the form of Insert, update, delete

TFD	PFD	TRFD
$A \rightarrow B$	$A \rightarrow C$	$A \rightarrow B$
$A \rightarrow C$	$B \rightarrow D$	$A \rightarrow C$
$A \rightarrow D$	$B \rightarrow C$	$C \rightarrow D$
Redundancy = No	Yes	Yes
Anomalies = No	Yes	Yes

• NORMALISATION :-

"Process of reducing a bigger table into smaller table by identifying redundancy & anomalies".

OR

"It is a process of reducing bigger table into smaller table by identifying repetations of side effects."

TYPES OF NORMALISATION :-

- 1) 1NF
- 2) 2NF
- 3) 3NF

1) 1NF

- Data entered in a cell should be single value data as atomic data.
- if the table is unconditional, convert into "total functional dependancies".

2) 2NF

- if the table is under 2NF first it should be complete with 1NF.
- if the table is having "Partial functional dependancy" it should convert into "total functional dependancy".

example:- consider we are having table big as;

Empno.	Ename	Deptno	Job	Deptno	Deptname	D-loc.

we will divide the table in 2 i.e. Emp table & Dept table.;

Emp table - ①

Empno.	Ename	Deptno	Job

Dept table - ②

Deptno.	Dname	loc

i.e.

(K.A)

Emp { empno. , ename , Job , Deptno

Dept { Deptno , Dname , loc

(K.A) (N.KA) (N.KA)

3) 3 NF

- if the table is under 3NF it should complete 2NF
- if the table is having transitive functional dependancies convert & make into total functional dependancy.

example :—

consider we are having as big table as follows;

empno.	ename	Job	Deptno	Dname	pincode	city	state.

Now we will divide the table in 3 i.e. Emp table & Dept table, pincode table.

Emp table → ①

empno.	ename	Job	Deptno

T.F.D (1NF)

Dept table → ②

Deptno	Dname	pincode

T.F.D (1NF)

pincode table → ③

pincode	city	state

T.F.D (1NF)

i.e.

Emp { empno, ename, Job, Deptno.
Dept { Deptno, Dname, pincode
Loc { pincode, city, state.

} 3 different tables will be created to avoid redundancy & anomalies.

DISADVANTAGE OF NORMALISATION :-

The only minor disadvantage is that we have to write complex queries as we have more no. of tables to be accessed.

- **DENORMALISATION :-**

"It is the process of combining more than 1 smaller table to form a bigger table. It is known as 'denormalization'.

- **CORRELATED SUBQUERY :-**

- when subquery is dependent on o/p of the outer query then we call it as "CORRELATED SUBQUERY".

- correlated subquery works on the principle of subquery & joins.

- The o/p of the outer query will be passed to subquery then the output of subquery will be passed to outer query & display the result for the user.