Normalization

Normalization -- Normalization is a process of organizing our data. It is a multi-step process that sets the data into tabular form and removes the duplicated data from the relational tables.

OR

Normalization is a technique to remove or reduce redundancy from a table. Redundancy means **having multiple copies of same data in the database**. This problem arises when a database is not normalized.

They are of 3 Types :--

(1.) First Normalization Form (1NF) --> Table should not contain any multivalued attributee.

Table --> This table is not in one normal form because student course has more than one value as it should only have one value .

sID	sName	sCourse
1	Amit	Spring Boot/Angular
2	Yash	Java/React
3	Nitin	Java

sID	sName	sCourse1	sCourse2
1	Amit	Spring Boot	Angular
2	Yash	Java	React
3	Nitin	Java	null

sID	sName
1	Amit
2	Yash
3	Nitin

sID	sCourse
1	Spring Boot
1	Angular
2	Java
2	React
3	Java

```
create database studentdata;
```

```
use studentdata;
```

```
create table student(sId int primary key auto_increment,sName varchar(20));
```

```
insert into student(sName) value("Amit");
insert into student(sName) value("Yash");
insert into student(sName) value("Nitin");

desc student;
select * from student;

create table course(sId int,sCourse varchar(20));
desc course;
select * from course;

alter table course add foreign key (sId) references student(sId);

insert into course values(1,"Spring Boot");
insert into course values(1,"Angular");
insert into course values(2,"Java");
insert into course values(2,"React");
insert into course values(3,"Java");

select * from student inner join course on student.sId = course.sId;

select student.sId,student.sName,course.sCourse from student inner join course on student.sId = course.sId;
```

(2.) Second Normalization Form (2NF) --> Table should be in 1NF. Table should not contain any partial dependency. All non-prime attributes should be fully functionally dependent on candidate key(composite primary key).
There should not be any partial dependency.
The Second Normal Form **eliminates partial dependencies on primary keys**.
Eg --> (AB) --> C ---------->2NF achieved
A --> C B--> C ----------> 2NF not achieved

Partial Dependency -- Partial dependency occurs when a part of composite key uniquely identifies a attribute in a table.

A	B	C
1	X	P
2	Y	Q
1	Z	R
3	Z	R
4	Z	R
5	Z	R

Relationship in this table :--
(AB) --> C ---------- As you can see composite key can uniquely identify a non key attribute.

B --> C ---------- But a part of composite key is also uniquely identifying a non key attribute and this creates redundancy or repetation of data.

(3.) Third Normalization Form (3NF) --> Table must be in 2NF and there should be no transitive dependency in the table.

Transitive Dependency --> When one column depends on a column which is not primary key.

OR Non-prime depends on non-prime.

Prime attribute --> prime attributes are primary keys on which all the other columns depend.

No non-prime should determine non-prime.

Eg --> X -> Y -> Z

Now, X is a prime attribute.

Y is a non-prime attribute. -----> X determines Y. because X is prime(primary key) and Y is non-prime.

Z is a non-prime attribute. -----> Y determines Z.