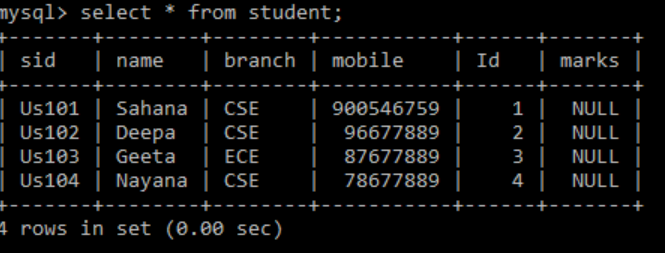
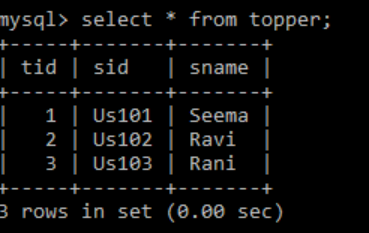
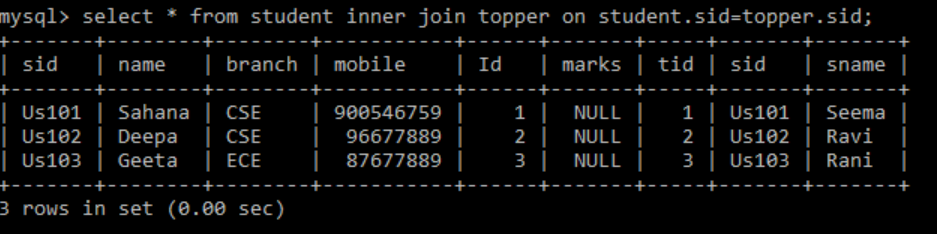
**SQL JOINS**

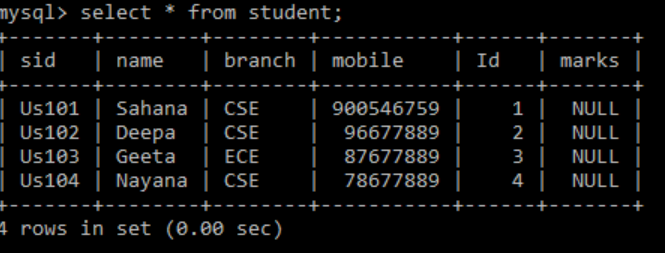
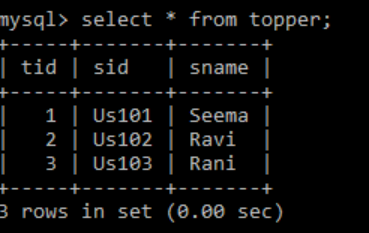
Join class is used to combine rows from two or more tables based on related columns between them.

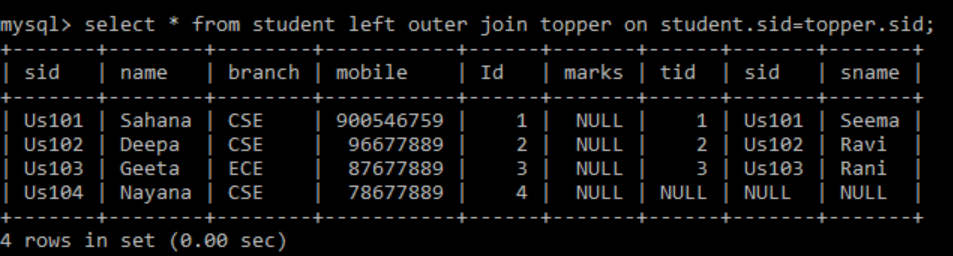
1. Inner Join: It returns records that have matching values in both tables.

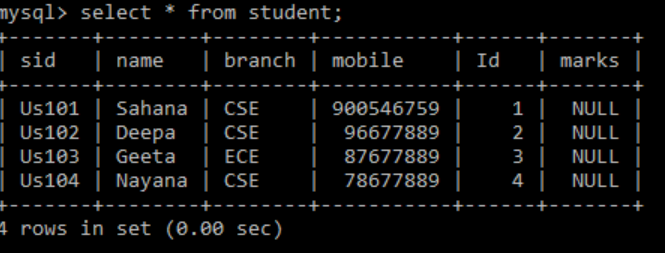
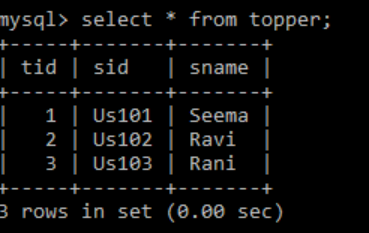


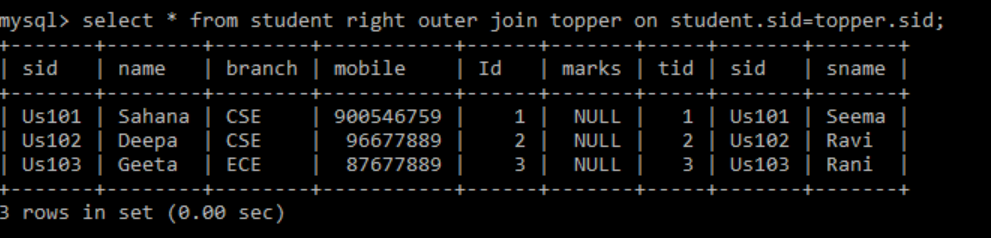
2. Left Outer Join: Returns all records from left table and matched record from right table.

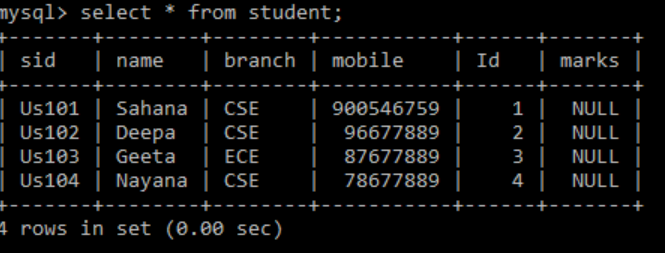
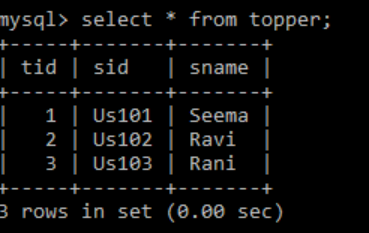


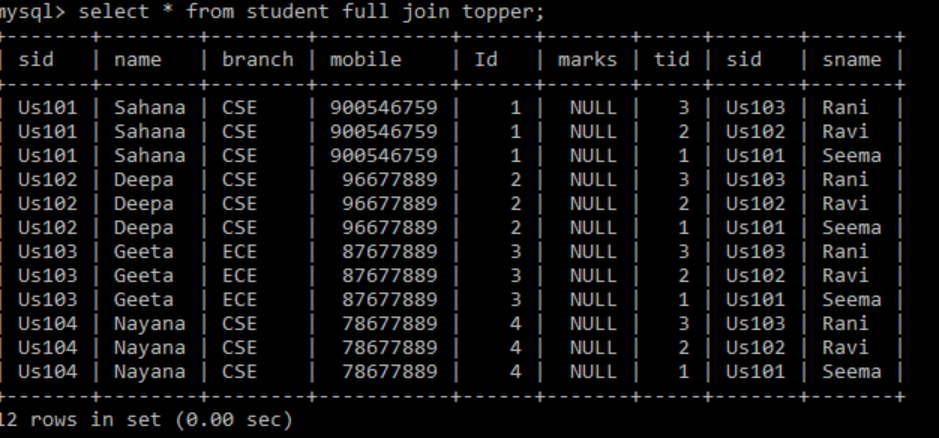
3. Right outer join: Returns all records from right table and matched record from left table.



4. Full outer join: Returns all records from left and right table and common between both tables.

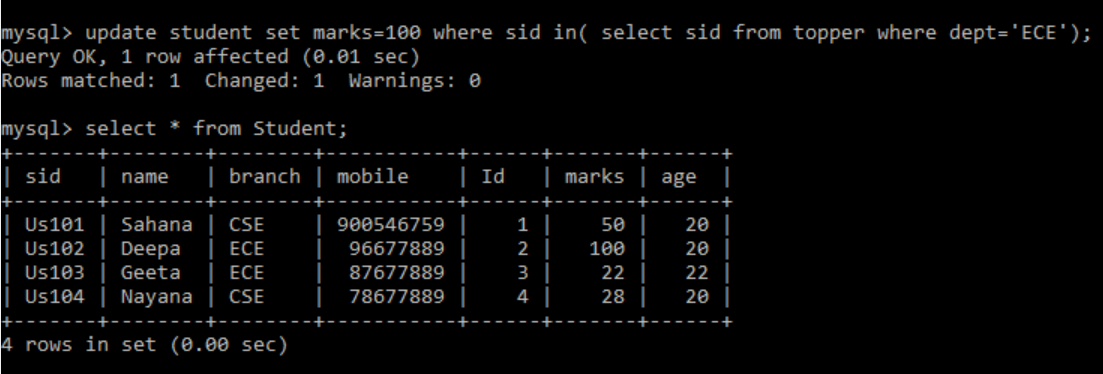
 



**SUB QUERY**

A sub query is a query within another query. A subquery is used to return a data that will be used in main query.

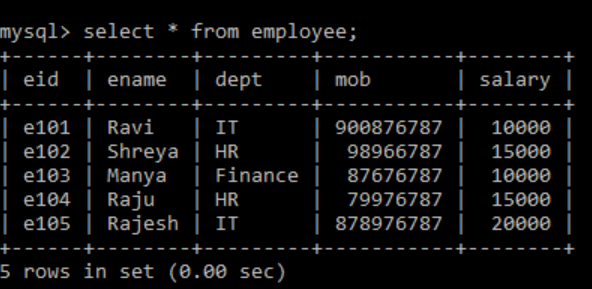
In this example the query is written to set the marks of student to 100, to the topper table from ECE department.



**Normalization:**

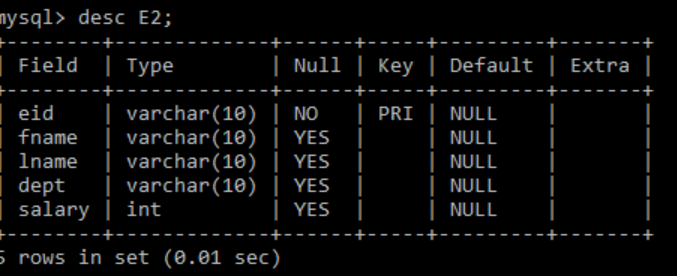
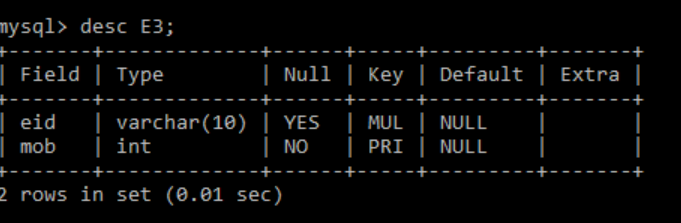
Process of reducing the redundancy of data in table also improving data integrity is normalization.

1. 1st Normal Form (1NF): The 1st normal form should be simple and atomic. It should not contain composite attribute and multivalued attribute.



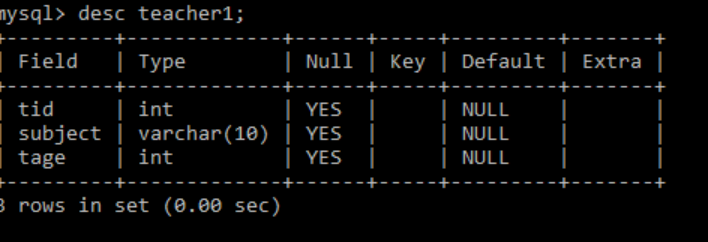
In this table ename (fname, lname) is composite attribute, and mobile is multivalued attribute because one employee can have multiple mobile number, so this is not in 1st normal form.

If we apply 1st normal form to above table we get:

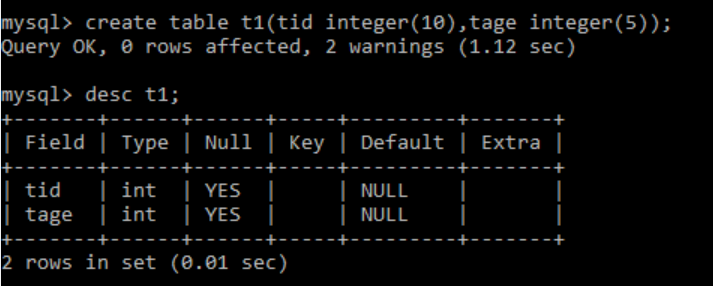
2. 2nd Normal form: The first condition in 2nd normal form is that the table has to be in 1st normal form.

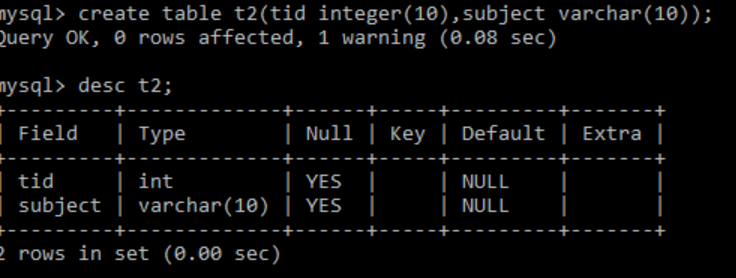
Consider an example:



In this table non- prime attribute tage is dependent on tid. So, this is not in 2nd normal form.

To convert into 2nd normal form, we divide the tables into two.

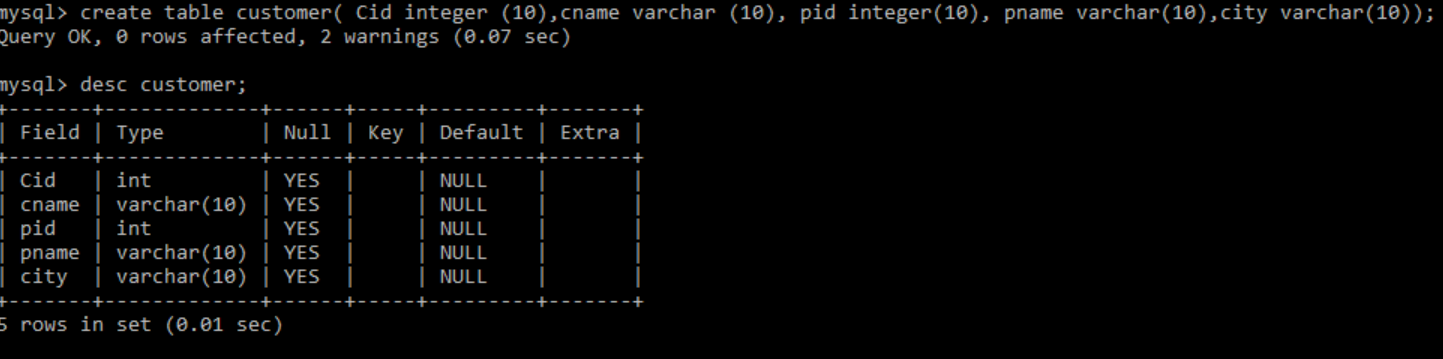




Now the tables are in 2nd Normal form.

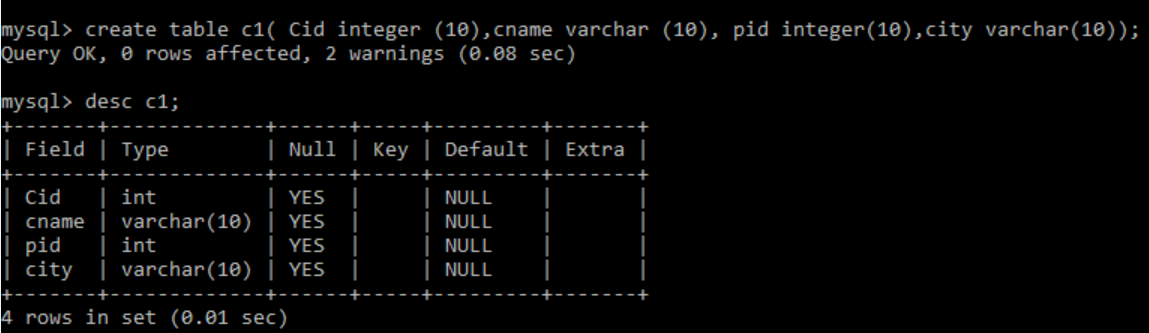
3. 3rd Normal Form: The first condition in 3nd normal form is that the table has to be in 2st normal form. The other condition is that there should be no transitive dependency for non-prime attributes.

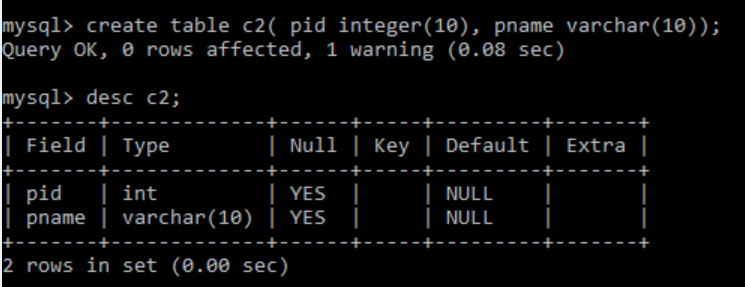
Ex:



In this example cid determines pid and pid determines pname. Hence, cid determines pname.

We have transitive dependency. So, in order to achieve 3NF we need to divide the table:





These are in 3NF as in first table c1 cname, pid and city are dependent on cid. And in second table c2 pname is only dependent on pid.