# **Assignment 8**

#### Q-15. Write an SQL query to fetch intersecting records of two tables.

**Answer:**

SELECT column1, column2, ... FROM table1 INTERSECT SELECT column1, column2, ... FROM table2;

For example, to retrieve the intersecting records from two tables, Employee and Manager, based on their respective salaries, the SQL query would be:

SELECT emp\_name, salary FROM Employee WHERE salary >= 50000 INTERSECT SELECT mgr\_name, salary FROM Manager WHERE salary >= 50000;

This query will return the names and salaries of employees and managers who have a salary greater than or equal to 50000 and are present in both Employee and Manager tables.

#### Q-16. Write an SQL query to show records from one table that another table does not have.

**Answer:**

To show records from one table that another table does not have, we can use the SQL LEFT JOIN operator with a WHERE clause. The LEFT JOIN operator returns all the rows from the left table (the first table mentioned in the query) and the matching rows from the right table (the second table mentioned in the query). If there is no matching row in the right table, then the result will contain NULL values for the columns of the right table. We can use this property of the LEFT JOIN operator to find the records that are present in one table but not in another.

The syntax for using LEFT JOIN operator is as follows:

SELECT column1, column2, ... FROM table1 LEFT JOIN table2 ON table1.column = table2.column WHERE table2.column IS NULL;

For example, to retrieve the names of employees from the Employee table who have not been assigned a manager in the Manager table, we can use the following SQL query:

SELECT emp\_name FROM Employee LEFT JOIN Manager ON Employee.emp\_id = Manager.emp\_id WHERE Manager.emp\_id IS NULL;

This query will return the names of all employees in the Employee table whose emp\_id is not present in the Manager table, indicating that they have not been assigned a manager yet.