
Experiment No. : 6

Title: Implementation of set operation and different types of joins.

Objectives:

1. To learn SQL set operations
2. To learn about inner join
3. To learn about left outer join
4. To learn about right outer join
5. To learn about full outer join.

Key Concepts: set operations, Inner join, left outer join, right outer join, full outer join

Theory:

Set operations

SET operators are mainly used to combine the same type of data from two or more tables. Although more than one select statement will then be present, only one result set is returned.

Rules on Set Operations:

- The result sets of all queries must have the same number of columns.
- In every result set the data type of each column must match the data type of its corresponding column in the first result set.
- In order to sort the result, an ORDER BY clause should be part of the last statement.
- The records from the top query must match the positional ordering of the records from the bottom query.
- The column names or aliases must be found out by the first select statement.

Four Set Operators:

The four set operators union, union all, intersect and except allow us to serially combine two or more select statements

Syntax :

(Query 1) UNION/ UNION ALL/ INTERSECT/ MINUS (Query 2);

UNION	The SQL Union Operator is used to combine two tables using select statement when both tables have the same number of columns. Union works like Distinct
UNION ALL	The SQL Union ALL Operator is used to combine two tables using select statement when both tables have the same number of columns. Union all DOES NOT do distinct
INTERSECT	INTERSECT returns any distinct values that are returned by both the query on the left and right sides of the INTERSECT operand
MINUS	MINUS query returns all rows which are in the first query but those are not returned in the second query.

JOINS

Join is one of the fundamental units of relational database system. Generally Join is used to combine related data from many tables to produce result.

There are different types of joins available,

- Inner Join
- Left Outer Join
- Right Outer Join
- Full Outer Join
- Cross Join
- Natural Join

Inner Join:

The default join is inner join. Inner join returns all matching records from the given datasets. The match is identified using predicates.

Select * from student INNER JOIN result ON student.student_id = result.student_id

Outer Join

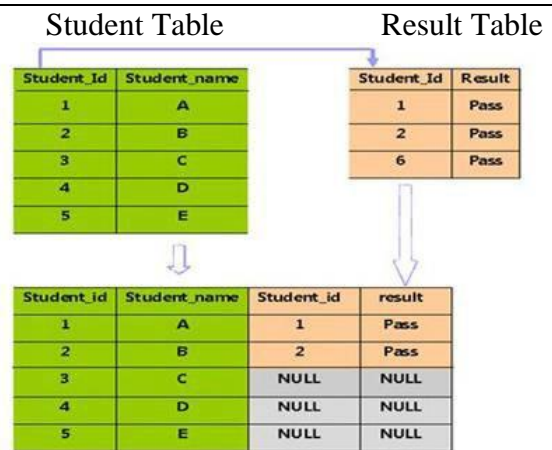
Outer join can be classified into Left outer join, Right outer join and Full outer join.

Left outer join:

Left outer join returns all records from left table, at the same time, it brings all matching rows from right table for the given predicate and NULL appears in right side when no matching row exists.

Left outer join can also be defined as “result set of inner join + missed records from left table with NULL in the right hand side”

select * from student LEFT OUTER JOIN result ON student.student_id = result.student_id

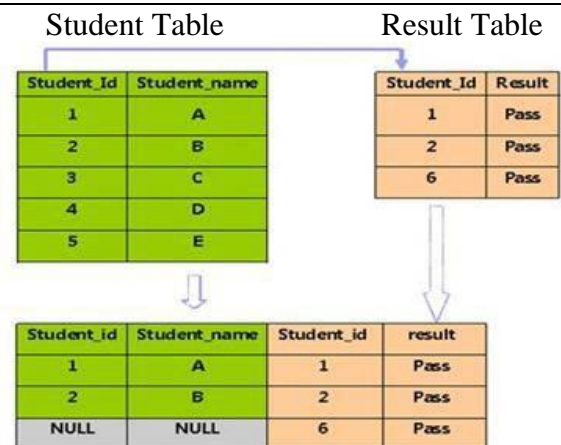


Right outer join:

Right outer join returns all records from right table, at the same time, it brings all matching rows from left table for the given predicate, and NULL appears in left side when no matching row exists.

Right outer join can also be defined as “result set of inner join + missed records from right table with NULL in the left hand side”

select * from student RIGHT OUTER JOIN result ON student.student_id = result.student_id



Full outer join:

A full outer join returns all rows in both the left and right tables. Any time a row has no match in the other table, such columns from the other table contain null values. When there is a match between the tables, the entire result set row contains data values from the base tables.

Select * from student FULL OUTER JOIN result ON student.student_id = result.student_id

Cross join

In Cross join every row of one table is matched with every row of another table. Cartesian join and cross join are same. If T1 and T2 are two sets then $T1 \times T2$ return $T1 * T2$ rows.

```
SELECT * FROM EMP CROSS JOIN DEPT;
```

```
SELECT * FROM EMP, DEPT;
```

Natural join

Joins two tables based on same values for same attribute in table..

```
SELECT * FROM STUDENT NATURAL JOIN RESULT;
```

LAB WORK:

Q1. Perform following queries on bank database:

1. Find the names and loan numbers of all customers who have a loan at the Perryridge branch.
2. Display the list in alphabetic order all customers who have a loan at the Perryridge branch.
3. Find names of all branches with customer who have account in the bank and live in city whose name start with S.
4. For all customers who have a loan from the bank, find their names, loan numbers, and loan amount
5. Find the list of all customers in alphabetic order who have a loan at the Parkstreet branch
6. For all customers who have a loan from the bank, find their names, loan numbers, and loan amount.
7. Find the customer names, loan numbers, and loan amounts, for all loans at the Perryridge branch.
8. For all customers who have a loan from the bank, find their names and loan numbers with the attribute loan_number replaced by loan_id.
9. Find the name, account number, and balance of all customers who have an account.
10. Find the name, loan number, and amount of all customers who have an loan from Perryridge branch
11. Find the name, account number, and balance of all customers who have an account with a balance of \$400 or less.
12. List all accounts of Brooklyn branch

Q2. Consider the following schema:

Sailors (sid, sname, rating, age)

Boats (bid, bname, color)

Reserves (sid, bid, day(date))

Write SQL statements for the following query.

1. Find all information of sailors who have reserved boat number 101.
2. Find the name of boat reserved by Bob.
3. Find the names of sailors who have reserved a red boat, and list in the order of age.

4. Find the names of sailors who have reserved at least one boat.
5. Find the ids and names of sailors who have reserved two different boats on the same day.
6. Find the ids of sailors who have reserved a red boat or a green boat.
7. Find the name and the age of the youngest sailor.
8. Count the number of different sailor names.
9. Find the average age of sailors for each rating level.
10. Find the average age of sailors for each rating level that has at least two sailors.