Assignment No. 4

- Title : To understand the use of joins, subquery and view with DML commands.

- Problem Statement : Design at least 10 SQL statements for suitable database applications using SQL DML statements (All types of joins, subqueries and views)

- Objectives : To,
1) Understand types of joins, subquery and view.
2) Understand how to use join with DML commands.
3) Perform updation on simple view.

- Outcomes : We will be able to,
1) Identify and implement types of join, subquery & views.
2) Implement and updation of simple view.

- S/W & H/W Requirements :
MySQL, PC with 64-bit OS Fedora / Windows 10, 8 GB RAM.

- Theory -

Joins -

SQL join is used to fetch data from two or more tables, which is joined to appear as single set of data. SQL Join is used for combining column from two or more tables by using values common to both tables. Join keyword is used in SQL queries for joining two or more tables.

Q. No.

Q. No.

Syntax: Select column_name() from table, right join table 2 on table1.columnname = table2.colname

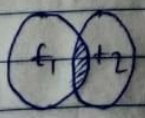
Types of Joins -

1) Cross Join:
This type of JOIN returns the cartesian product of rows from the tables in JOIN. It will return a table which consists of records which combines each row from the first table with each row of the second table.

Syntax:
Select column_name_list
From table_name1 CROSS JOIN table_name2

2) Inner Join:
This is a simple join in which the result is based on matched data as per the equality condition specified in the query.

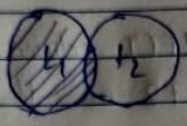
Syntax:
SELECT column_name_list from table_name1 INNER JOIN table_name2 WHERE table_name1 column_name = table_name column_name;



3) Outer JOIN:

Outer join is based on both matched to unmatched data. Outer joins subdivide into three:

a) left b) right c) full



a) Left outer join:

The left outer join returns a result table with the matched data of two tables then remaining rows of the left table and null for the right table's column.

Syntax:

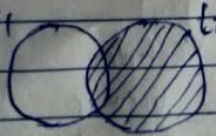
SELECT column_name_list from table_name1 LEFT OUTER JOIN table_name2 on table_name1.columnname = tablename2.colname

b) Right Outer join:

The right outer join returns a result table with the matched data of two tables then remaining rows of the right table and null for the left table's column.

Syntax -

SELECT column_name_list from t1 RIGHT OUTER JOIN t2 on t1.colname = t2.colname



Sr. No.

c) Full outer join:

The full outer join returns a result table with the matched data of two table then the remaining rows of both left table and then right table.

Syntax: SELECT column_name_list From t1 FULL OUTER JOIN t2 on t1.colname = t2.colname

- Test Cases:

	Input	Expected o/p	Actual O/p	Result
1)	Select * from cust natural join author	Error (No common Field)	Error	Success
2)	Select * From cust natural join orders	display all order details	display of details of orders	Success
3)	Create view V1 as select * from cust	Error (Already existing)	error	Success
4)	Create view V1 as select * from cust natural join order	complex view is created	complex view is created	Success

Conclusion

Hence, we successfully implemented queries using various type of join, concepts of creation of complex view and simple view