Assignment A4

Date:

TITLE	Design at least 10 SQL queries for suitable database application using SQL DML statements: All types of join, sub-query and View.
PROBLEM STATEMENT / DEFINITION	Design at least 10 SQL queries for suitable database application using SQL DML statements: All types of join, sub-query and View.
LEARNING OBJECTIVE	• To understand types of join, subquery and view • To understand how to use join with DML commands • To perform updates on simple view
LEARNING OUTCOME	The students will be able to • Identify and implement types of join, subquery and view • Implementation and updating of simple view
S/W PACKAGES & HARDWARE APPARATUS USED	• MySQL • 64-bit Linux based open source OS • 8 GB RAM

CONCEPT RELATED THEORY:

JOIN:

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. Minimum required condition for joining table, is (n-1) where n, is number of tables. A table can also join to itself known as, Self-Join.

Types of join:

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;

Inner join: This is a simple JOIN in which 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-name2.column-name;

Natural join: Natural Join is a type of Inner join which is based on column having same name and same datatype present in both the tables to be joined.

Syntax: SELECT * from table-name1 NATURAL JOIN table-name2;

Outer join: Outer join is based on both matched and unmatched data. Outer joins subdivide further into,

- Left Outer Join

- Right Outer Join
- Full Outer Join

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.column-name = table-name2.column-name;

Right Outer Join: The right outer join results a result table with the matched data of two tables then remaining rows of the right table and null for the left table’s columns

Syntax: SELECT column-name-list from table-name1 RIGHT OUTER JOIN table-name2 on table-name1.column-name = table-name2.column-name;

Full Outer Join: The full outer join results a result table with the matched data of two table then remaining rows of both left table and then the right table.

Syntax: SELECT column-name-list from table-name1 FULL OUTER JOIN table-name2 on table-name1.column-name = table-name2.column-name;

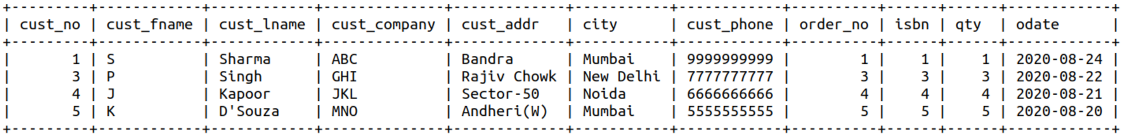
TEST CASES:

Description:

1. Find Customer details and order details using NATURAL JOIN.
2. Find the book_ title, author_name, country.
3. Find the customer ID, name and order_no of customers who have never placed an order.
4. Find the Title, ISBN, order_no of the books for which order is not placed.
5. Display cust_fname, title,author_no,publisher_year where ISBN=5.
6. Display the total number of books and customer name.
7. List the cust_id, order_no and ISBN with books having title 'hello'.
8. Find the names of all the companies that ordered books in the year 2015.
9. Create view showing the author and book details.
10. Perform Manipulation on simple view-Insert, update, delete, drop view.

Input/Output:

```
1. Select * from customer natural join orders;
```



```
+---------+------------+------------+--------------+-------------+------------+------------+----------+------+-----+------------+
| cust_no | cust_fname | cust_lname | cust_company | cust_addr   | city       | cust_phone | order_no | isbn | qty | odate      |
+---------+------------+------------+--------------+-------------+------------+------------+----------+------+-----+------------+
|       1 | S          | Sharma     | ABC          | Bandra      | Mumbai     | 9999999999 |        1 |    1 |   1 | 2020-08-24 |
|       3 | P          | Singh      | GHI          | Rajiv Chowk | New Delhi  | 7777777777 |        3 |    3 |   3 | 2020-08-22 |
|       4 | J          | Kapoor     | JKL          | Sector-50   | Noida      | 6666666666 |        4 |    4 |   4 | 2020-08-21 |
|       5 | K          | D'Souza    | MNO          | Andheri(W)  | Mumbai     | 5555555555 |        5 |    5 |   5 | 2020-08-20 |
+---------+------------+------------+--------------+-------------+------------+------------+----------+------+-----+------------+
```

```
2. select book.title, author.author_name, author.country from
   book inner join author on book.author_no = author.author_no;

+-------+-------------+---------+
| title | author_name | country |
+-------+-------------+---------+
| Hello | KO          | India   |
| Bye   | JKR         | UK      |
| Boom  | IA          | Russia  |
+-------+-------------+---------+
3 rows in set (0.00 sec)
```

3. select cust_no, cust_fname, cust_lname from customer where not exists (select order_no from orders where customer.cust_no = orders.cust_no);

```
+---------+------------+------------+
| cust_no | cust_fname | cust_lname |
+---------+------------+------------+
|       7 | S          | Holmes     |
|       8 | T          | Stark      |
|       9 | E          | Macron     |
|      10 | H          | Lesnitsky  |
+---------+------------+------------+
4 rows in set (0.00 sec)
```

4. select book.title, book.isbn, orders.order_no from book inner join orders on book.isbn = orders.isbn where orders.order_no = null;
   Empty set (0.00 sec)

5. select customer.cust_fname, book.title, book.author_no, book.publisher_no from customer inner join orders on customer.cust_no = orders.cust_no inner join book on orders.isbn = book.isbn where book.isbn = 5;

```
+------------+--------+------------+--------------+
| cust_fname | title  | author_no  | publisher_no |
+------------+--------+------------+--------------+
| K          | Thanks | 5555555555 | 5050505050   |
+------------+--------+------------+--------------+
1 row in set (0.00 sec)
```

6. select customer.cust_fname, count(*) as book_count from book inner join orders on book.isbn = orders.isbn inner join customer on customer.cust_no = orders.cust_no group by customer.cust_fname;

```
+------------+------------+
| cust_fname | book_count |
+------------+------------+
| S          |          1 |
| P          |          1 |
| J          |          1 |
| K          |          1 |
+------------+------------+
4 rows in set (0.00 sec)
```

7. select customer.cust_no, orders.order_no, orders.isbn from customer inner join orders on customer.cust_no = orders.order_no inner join book on orders.isbn = book.isbn where book.title = 'hello';

```
+---------+----------+------+
| cust_no | order_no | isbn |
+---------+----------+------+
|       1 |        1 |    1 |
+---------+---------+------+
1 row in set (0.00 sec)
```

```
8. select customer.cust_company from customer inner join orders
   on customer.cust_no = orders.cust_no inner join book on
   orders.isbn = book.isbn where book.publisher_year = '2015';
```

```
+-------------+
| cust_company |
+--------------+
| GHI          |
+--------------+
1 row in set (0.00 sec)
```

```
9. create view author_book as select * from book natural join
   author;
Query OK, 0 rows affected (0.47 sec)
```

```
select * from author_book;
+------------+------+-------+------------+--------------+----------------+-------------+---------+
| author_no  | isbn | title | unit_price | publisher_no | publisher_year | author_name | country |
+------------+------+-------+------------+--------------+----------------+-------------+---------+
| 9999999999 |    1 | Hello |        123 | 1111111111   |           2006 | KO          | India   |
| 7777777777 |    3 | Bye   |        210 | 3333333333   |           2015 | JKR         | UK      |
| 6666666666 |    4 | Boom  |        340 | 4444444444   |           2014 | IA          | Russia  |
+------------+------+-------+------------+--------------+----------------+-------------+---------+
3 rows in set (0.00 sec)
```

```
10. alter view author_book as select book.title, author.country
from book natural join author;
Query OK, 0 rows affected (0.77 sec)
```

```
mysql> select * from author_book;
+-------+---------+
| title | country |
+-------+---------+
| Hello | India   |
| Bye   | UK      |
| Boom  | Russia  |
+-------+---------+
3 rows in set (0.00 sec)
```

```
mysql> drop view author_book;
Query OK, 0 rows affected (0.89 sec)
```