

Database Systems Lab - 14CS2012

REGISTER NO: UR14CS228

DATE: 26-08-16

EXPERIMENT-NO 5

Video Link : <https://youtu.be/RsiWTjJedfI>

AIM: To write Basic SQL queries, Sub Queries and Views.

DESCRIPTION:

Subqueries can be used with the SELECT, INSERT, UPDATE, and DELETE statements along with the operators like =, <, >, >=, <=, IN, BETWEEN etc.

A subquery is used to return data that will be used in the main query as a condition to further restrict the data to be retrieved.

A Subquery or Inner query or Nested query is a query within another SQL query and embedded within the WHERE clause.

Output:

1.Find the names of customers who have ordered for a `cherry' or `red oak' finish.

```
SELECT * FROM Customer WHERE customer_id IN (SELECT customer_id FROM
Orders WHERE p_id IN (SELECT product_id FROM Product WHERE
product_finish='cherry' OR product_finish='red oak'))
```

NO ROWS SELECTED

2.Find the names of customers who have ordered for the product "Table Lamp". (use IN)

```
SELECT * FROM Customer WHERE customer_id IN (SELECT customer_id FROM
Orders WHERE p_id IN (SELECT product_id FROM Product WHERE
product_description='Table Lamp'))
```

NO ROWS SELECTED

3.Find the details of customers who have ordered for product id 1000. (use EXISTS)

```
SELECT * FROM Customer WHERE customer_id IN (SELECT customer_id FROM
Orders WHERE p_id='1000')
```

CUSTOMER_ID	CUSTOMER_NAME	CUSTOMER_ADDRESS	CITY	STATE	PSTAL
2	Mary Smith	6900 Main St.	SanFrancis	CA	10032

4.Find suppliers whose total supply quantity is greater than some supplier called S3. (use ANY)

```
SELECT * FROM Supplier WHERE supply_quantity > (SELECT supply_quantity
FROM Supplier WHERE s_id ='S3')
```

NO ROWS SELECTED

5.Find the supplier with the highest rating using ALL

```
SELECT s_name FROM Supplier WHERE p_id= ANY (SELECT product_id FROM
Orderquantity WHERE ordered_quantity= (SELECT MAX(ordered_quantity) FROM
Orderquantity))
```

S_NAME

aaa

6.Display the products that are same as the product ordered by "Cathy Cook"

```
SELECT Product.* FROM Product WHERE product_id = ANY (SELECT p_id FROM
Order WHERE customer_id = (SELECT customer_id FROM Customer WHERE
customer_name ='Cathy Cook'))
```

PRODUCT_ID PRODUCT_DESCRIPTION

3001 Duplex Book Shelf

7.Display the supplier details whose supplied quantity is greater than the quantity of order id - 103

```
SELECT * FROM Supplier WHERE s_id = ANY (SELECT s_id FROM Orders
WHERE ordered_quantity > (SELECT ordered_quantity FROM Orderqunantity WHERE
order_id='103'))
```

S_ID	S_NAME	GENDER	S_DATE	P_ID
------	--------	--------	--------	------

1	aaa	M	11-NOV-14	1000
---	-----	---	-----------	------

8.Display the ordered quantity of the order whose ordered date is greater than the ordered date of the order id 106

```
SELECT ordered_quantity FROM Orderquantity WHERE order_id = ANY (SELECT
order_id FROM Orders WHERE order_date > (SELECT order_date FROM Orderquantity
WHERE order_id='106') )
```

NO ROWS SELECTED

9.Display the customer names whose order date is equal to the order date of customer id 4.

```
SELECT * FROM Customer WHERE customer_id = ANY (SELECT customer_id
FROM Orders WHERE order_date > ANY (SELECT order_date FROM Orders WHERE
customer_id='4'))
```

CUSTOMER_NAME

Cathy Cook

Mary Smith

John Doe

10.Display the customer details whose rating is greater than the supplier "ccc"

```
SELECT * FROM Customer WHERE customer_id = ANY (SELECT rating FROM
Supplirt WHERE s_name='ccc')
```

NO ROWS SELECTED

11.Create a view(V1) to display the customer name, order id and quantity.

```
CREATE OR REPLACE VIEW V1 AS (SELECT
customer_name,Orders.order_id,Orderquantity.ordered_quantity FROM Customer INNER
JOIN Orders ON Customer.customer_id=Orders.customer_id INNER JOIN Orderquantity
ON Orderquantity.order_id=Orders.order_id);
```

View created.

12.Create a view(V2) to display the order-id, order-date and the date of delivery.

```
CREATE OR REPLACE VIEW V2 AS (SELECT  
Orders.order_id,Orders.order_date,Supplier.s_date FROM Orders INNER JOIN  
Orderquantity ON Orderquantity.order_id=Orders.order_id INNER JOIN Supplier ON  
Supplier.p_id=Orderquantity.product_id);
```

View created.

13.Insert a row into the above view. Is the view updatable? If yes, is the updation reflected in the base table.

```
INSERT INTO V2 VALUES (104, TO_DATE('10-OCT-2014','DD-MON-YYYY'),  
TO_DATE('15-OCT-2014','DD-MON-YYYY'));
```

View Updated

14.Create a view (V3) with customer name, order id and order date from V1 and V2.

```
CREATE OR REPLACE VIEW V3 AS (SELECT  
V1.customer_name,V1.order_id,V2.order_date FROM V1 INNER JOIN V2 ON  
V1.order_id=V2.order_id);
```

View created.

15.Delete a row from the above view and state what happens to the other views from which it was created.

```
DELETE FROM v3 WHERE V3.order_id=103;  
DELETE FROM v3 *
```

View deleted

Result:

SQL queries using DML and TCL commands are successfully executed.