**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.

**Program:**

**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.