Exp. No : 5a	**VIEWS**
Date :	

2360322 - Alan Chris Disilva

AIM:

To implement the Commands of Views for application in DBMS.

Question:

1. Write a query to create a view named CustomerCityView that includes CustomerID, CustomerName, and City from the Customer table.
2. Write a query to create a view named CustomerOrderSummary that combines data from Customer and OrderDetails. Include the columns CustomerName, City, OrderedItem, and OrderAmount.
3. Modify the CustomerCityView view to include the ContactNumber column. Use the CREATE OR REPLACE VIEW statement.
4. Create a view named ChennaiOrdersView that displays all orders placed by customers from Chennai. Use a join between Customer and OrderDetails.
5. Create a view named ElectronicsOrders that displays all orders where the OrderedItem is "Electronics."
6. Create a view named CustomerOrdersView that includes CustomerID and OrderAmount. Write a query to delete an order from the OrderDetails table using this view.
7. Write a query to drop the CustomerCityView view from the database.

Solution:

SQL> connect

Enter user-name: system

Enter password:

Connected.

SQL> create table customer(customer_id int, customer_name varchar(15), city varchar(15), contact_number number(10));

Table created.

SQL> insert into customer values(1, 'Alice', 'chennai', 9876543210);

1 row created.

SQL> insert into customer values(2, 'Bob', 'Bangalore', 8765432109);

1 row created.

SQL> insert into customer values(3, 'Charlie', 'Mumbai', 7654321098);

1 row created.

2360322 - Alan Chris Disilva

SQL> insert into customer values(4, 'Diana', 'Kolkata', 6543210987);

1 row created.

SQL> create table order_details(order_id int, customer_id int, order_amount int , ordered_item varchar(15));

Table created.

SQL> insert into order_details values(101, 1, 500.00, 'Books');

1 row created.

SQL> insert into order_details values(102, 2, 1000.00, 'Electronics');

1 row created.

SQL> insert into order_details values(103, 3, 750.00, 'Kitchen Appl');

1 row created.

SQL> insert into order_details values(104, 4, 1200.00, 'Furniture');

1 row created.

SQL> select * from customer;

CUSTOMER_ID CUSTOMER_NAME CITY CONTACT_NUMBER

---------- --------------- --------------- --------------

1 Alice chennai 9876543210

2 Bob Bangalore 8765432109

3 Charlie Mumbai 7654321098

4 Diana Kolkata 6543210987

SQL> select * from order_details;

ORDER_ID CUSTOMER_ID ORDER_AMOUNT ORDERED_ITEM

---------- ---------- ------------ ---------------

101 1 500 Books

102 2 1000 Electronics

103 3 750 Kitchen Appl

104 4 1200 Furniture

SQL> drop view customercityview ;

View dropped.

SQL> create view CustomerCityView as select customer_id, customer_name, city from customer;

View created.

SQL> drop view CustomerOrderSummary ;

View dropped.

2360322 - Alan Chris Disilva

SQL> create view CustomerOrderSummary as select c.customer_name,c.city, o.ordered_item, o.order_amount from customer c join order_details o on c.customer_id =o.customer_id;

View created.

SQL> create or replace view CustomerCityView as select customer_id, customer_name, city, contact_number from customer;

View created.

SQL> select order_id, customer_id, order_amount, ordered_item from order_details where ordered_item = 'Electronics';

ORDER_ID CUSTOMER_ID ORDER_AMOUNT ORDERED_ITEM

---------- ---------- ---------- ----------

102 2 1000 Electronics

SQL> create view customerorderview as select customer_id, order_amount from order_details;

View created.

SQL> delete from customerorderview where order_amount = 1000.00;

1 row deleted.

SQL> drop view customercityview;

View dropped.

SQL>

Result:

Thus, the implementation of View commands for applications in DBMS has been successfully executed and verified