OLUWATOSIN NGOBOH

COHOT 7 DATA SCIENCE

-- DATA QUERY LANGUAGE

-- Utilizing the provided tables - Products, Customers, Orders, OrderDetails, and ProductTypes -

the objective is to formulate SQL queries to retrieve relevant data using Data Query Language (DQL).

```
CREATE DATABASE "Data Query Project"
USE Data Query Project

-- creating a table called Products
CREATE TABLE Products(
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR(100),
    ProductType VARCHAR(50),
    Price DECIMAL(8, 2),
);
INSERT INTO Products(ProductID, ProductName, ProductType, Price)
VALUES (1, 'Widget A', 'Widget', '10.00'),
    (2, 'Widget B', 'Widget', '15.00'),
    (3, 'Gadget X', 'Gadget', '20.00'),
    (4, 'Gadget Y', 'Gadget', '25.00'),
    (5, 'Doohickey Z', 'Doohickey', '30.00');

SELECT * FROM Products



-- creating a table called Customers
CREATE TABLE Customers(
    CustomerID INT PRIMARY KEY,
    CustomerName VARCHAR(100),
    Email VARCHAR(30),
    Phone VARCHAR(24) NOT NULL,
);
```

```
INSERT INTO Customers(CustomerID, CustomerName, Email, Phone)
VALUES (1, 'John Smith', 'john@example.com', '123-456-7890'),
      (2, 'Jane Doe', 'jane.doe@example.com', '987-654-3210'),
      (3, 'Alice Brown', 'alice.brown@example.com', '456-789-0123');

   select * from Customers



-- creating a table called Orders_1
CREATE TABLE Orders_1(
      OrderID INT PRIMARY KEY,
      CustomerID INT,
      OrderDate DATE,
);
INSERT INTO Orders_1(OrderID, CustomerID, OrderDate)
VALUES ('101', 1, '2024-05-01'),
      ('102', 2, '2024-05-02'),
      ('103', 3, '2024-05-01');




-- creating a table called OrderDetails
CREATE TABLE OrderDetails(
       OrderDetailID INT PRIMARY KEY,
       OrderID INT,
       ProductID INT,
       Quantity INT,
);



INSERT INTO OrderDetails(OrderDetailID, OrderID, ProductID, Quantity)
VALUES (1, '101', 1, 2),
      (2, '101', 3, 1),
      (3, '102', 2, 3),
      (4, '102', 4, 2),
      (5, '103', 5, 1);
```

```
-- creating a table called ProductTypes
CREATE TABLE ProductTypes(
        ProductTypeID INT PRIMARY KEY,
        ProductTypeName VARCHAR(100),
);

INSERT INTO ProductTypes(ProductTypeID, ProductTypeName)
VALUES (1, 'Widget'),
       (2, 'Gadget'),
       (3, 'Doohickey');

       -- RETRIEVING ALL 1-5
SELECT * FROM Products, Customers, Orders_1, OrderDetails, ProductTypes

--6. RETRIEVING AS INSTRUCTED
SELECT * FROM Products;
SELECT * FROM Customers;
SELECT * FROM Orders_1;
SELECT * FROM OrderDetails;
SELECT * FROM ProductTypes;

--7. Retrieve the names of the products that have been ordered by at least one customer,

SELECT
P.ProductName,
    SUM(OD.Quantity) AS TotalQuantityOrdered
FROM
    Products P
LEFT JOIN
    OrderDetails OD ON P.ProductID = OD.ProductID
LEFT JOIN
    Orders_1 O ON OD.OrderID = O.OrderID
GROUP BY
    P.ProductName;
--
--8. Retrieve the names of the customers who have placed the most orders, along with the
```

total number of orders placed by each customer.

```
SELECT
    C.CustomerName,
    COUNT(O.OrderID) AS TotalOrders
FROM
    Customers C
LEFT JOIN
    Orders_1 O ON C.CustomerID = O.CustomerID
GROUP BY
    C.CustomerName
HAVING
    COUNT(DISTINCT O.OrderDate) = 7;

-- Assuming there are orders for every day of the week

--
SELECT
    C.CustomerName,
    COUNT(O.OrderID) AS TotalOrders
FROM
    Customers C
LEFT JOIN
    Orders_1 O ON C.CustomerID = O.CustomerID
GROUP BY
    C.CustomerName
ORDER BY
    TotalOrders DESC;
-- To get the customer with the most orders



SELECT
    P.ProductName,
    SUM(OD.Quantity) AS TotalQuantityOrdered
FROM
    Products P
JOIN
```

```
    OrderDetails OD ON P.ProductID = OD.ProductID
GROUP BY
    P.ProductName
ORDER BY
    TotalQuantityOrdered DESC;
```

```
SELECT DISTINCT
    C.CustomerName
FROM
    Customers C
JOIN
    Orders_1 O ON C.CustomerID = O.CustomerID
JOIN
    OrderDetails OD ON O.OrderID = OD.OrderID
JOIN
    Products P ON OD.ProductID = P.ProductID
WHERE
    P.ProductType = 'Widget';
```

```
SELECT
    C.CustomerName,
    SUM(P.Price * OD.Quantity) AS TotalCost
FROM
    Customers C
JOIN
    Orders_1 O ON C.CustomerID = O.CustomerID
JOIN
    OrderDetails OD ON O.OrderID = OD.OrderID
JOIN
    Products P ON OD.ProductID = P.ProductID
WHERE
    P.ProductType IN ('Widget', 'Gadget')
GROUP BY
    C.CustomerName
```

```
HAVING
   SUM(CASE WHEN P.ProductType = 'Widget' THEN OD.Quantity ELSE 0 END) > 0
   AND SUM(CASE WHEN P.ProductType = 'Gadget' THEN OD.Quantity ELSE 0 END) > 0;


SELECT
   C.CustomerName,
   SUM(P.Price * OD.Quantity) AS TotalCost
FROM
   Customers C
JOIN
   Orders_1 O ON C.CustomerID = O.CustomerID
JOIN
   OrderDetails OD ON O.OrderID = OD.OrderID
JOIN
   Products P ON OD.ProductID = P.ProductID
WHERE
   P.ProductType = 'Gadget'
GROUP BY
   C.CustomerName;

SELECT
   C.CustomerName,
   SUM(P.Price * OD.Quantity) AS TotalCost
FROM
   Customers C
JOIN
   Orders_1 O ON C.CustomerID = O.CustomerID
JOIN
   OrderDetails OD ON O.OrderID = OD.OrderID
JOIN
   Products P ON OD.ProductID = P.ProductID
WHERE
   P.ProductType = 'Doohickey'
GROUP BY
   C.CustomerName;
```