210030140
D BALAJI REDDY

Question:

For this example, let's assume we have a simple database for a retail store. Our database has the following tables:

1. Customers : Information about customers.

2. Products: Information about products.

3. Orders: Information about orders made by customers.

4. OrderItems: Information about items in each order.

Table Structures

Customers

CustomerID	FirstName	LastName	Email	DateOfBirth
1	John	Doe	john.doe@example.com	1985-01-15
2	Jane	Smith	jane.smith@example.com	1990-06-20

Products

ProductID	ProductName	Price
1	Laptop	1000
2	Smartphone	600
3	Headphones	100

Orders

OrderID	CustomerID	OrderDate
1	1	2023-01-10
2	2	2023-01-12

210030140
D BALAJI REDDY

OrderItems

OrderItemID	OrderID	ProductID	Quantity
1	1	1	1
2	1	3	2
3	2	2	1
4	2	3	1

Sample Queries

1. List all customers.

2. Find all orders placed in January 2023.

3. Get the details of each order, including the customer name and email.

4. List the products purchased in a specific order (e.g., OrderID = 1).

5. Calculate the total amount spent by each customer.

6. Find the most popular product (the one that has been ordered the most).

7. Get the total number of orders and the total sales amount for each month in 2023.

8.Find customers who have spent more than $1000.

Ans:

SQL Queries for Creation of tables:

```
CREATE TABLE Customers (
    CustomerID INT PRIMARY KEY,
    FirstName VARCHAR(50),
    LastName VARCHAR(50),
```

```
    Email VARCHAR(100),
    DateOfBirth DATE
);

CREATE TABLE Products (
    ProductID INT PRIMARY KEY,
    ProductName VARCHAR(100),
    Price DECIMAL(10, 2)
);

CREATE TABLE Orders (
    OrderID INT PRIMARY KEY,
    CustomerID INT,
    OrderDate DATE,
    FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
);

CREATE TABLE OrderItems (
    OrderItemID INT PRIMARY KEY,
    OrderID INT,
    ProductID INT,
    Quantity INT,
    FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
    FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
);
```

210030140
D BALAJI REDDY

SQL Queries for Insert records into schema:

```
INSERT INTO Customers (CustomerID, FirstName, LastName, Email, DateOfBirth) VALUES
(1, 'John', 'Doe', 'john.doe@example.com', '1985-01-15'),
(2, 'Jane', 'Smith', 'jane.smith@example.com', '1990-06-20');

INSERT INTO Products (ProductID, ProductName, Price) VALUES
(1, 'Laptop', 1000.00),
(2, 'Smartphone', 600.00),
(3, 'Headphones', 100.00);

INSERT INTO Orders (OrderID, CustomerID, OrderDate) VALUES
(1, 1, '2023-01-10'),
(2, 2, '2023-01-12');

INSERT INTO OrderItems (OrderItemID, OrderID, ProductID, Quantity) VALUES
(1, 1, 1, 1),
(2, 1, 3, 2),
(3, 2, 2, 1),
(4, 2, 3, 1);
```

210030140
D BALAJI REDDY

1. List all customers.

SELECT * FROM Customers;

2. Find all orders placed in January 2023.

SELECT * FROM Orders WHERE OrderDate BETWEEN '2023-01-01' AND '2023-01-31';

3. Get the details of each order, including the customer name and email.

SELECT Orders.OrderID, Customers.FirstName, Customers.LastName, Customers.Email, Orders.OrderDate FROM Orders JOIN Customers ON Orders.CustomerID = Customers.CustomerID;

4. List the products purchased in a specific order (e.g., OrderID = 1).

SELECT Products.ProductName, OrderItems.Quantity FROM OrderItems JOIN Products ON OrderItems.ProductID = Products.ProductID WHERE OrderItems.OrderID = 1;

5. Calculate the total amount spent by each customer.

SELECT Customers.CustomerID, Customers.FirstName, Customers.LastName, SUM(Products.Price * OrderItems.Quantity) AS TotalSpent FROM Customers JOIN Orders ON Customers.CustomerID = Orders.CustomerID JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID JOIN Products ON OrderItems.ProductID = Products.ProductID GROUP BY Customers.CustomerID, Customers.FirstName, Customers.LastName;

6. Find the most popular product (the one that has been ordered the most).

SELECT Products.ProductID, Products.ProductName, SUM(OrderItems.Quantity) AS TotalOrdered FROM OrderItems JOIN Products ON OrderItems.ProductID = Products.ProductID GROUP BY Products.ProductID, Products.ProductName ORDER BY TotalOrdered DESC LIMIT 1;

7. Get the total number of orders and the total sales amount for each month in 2023.

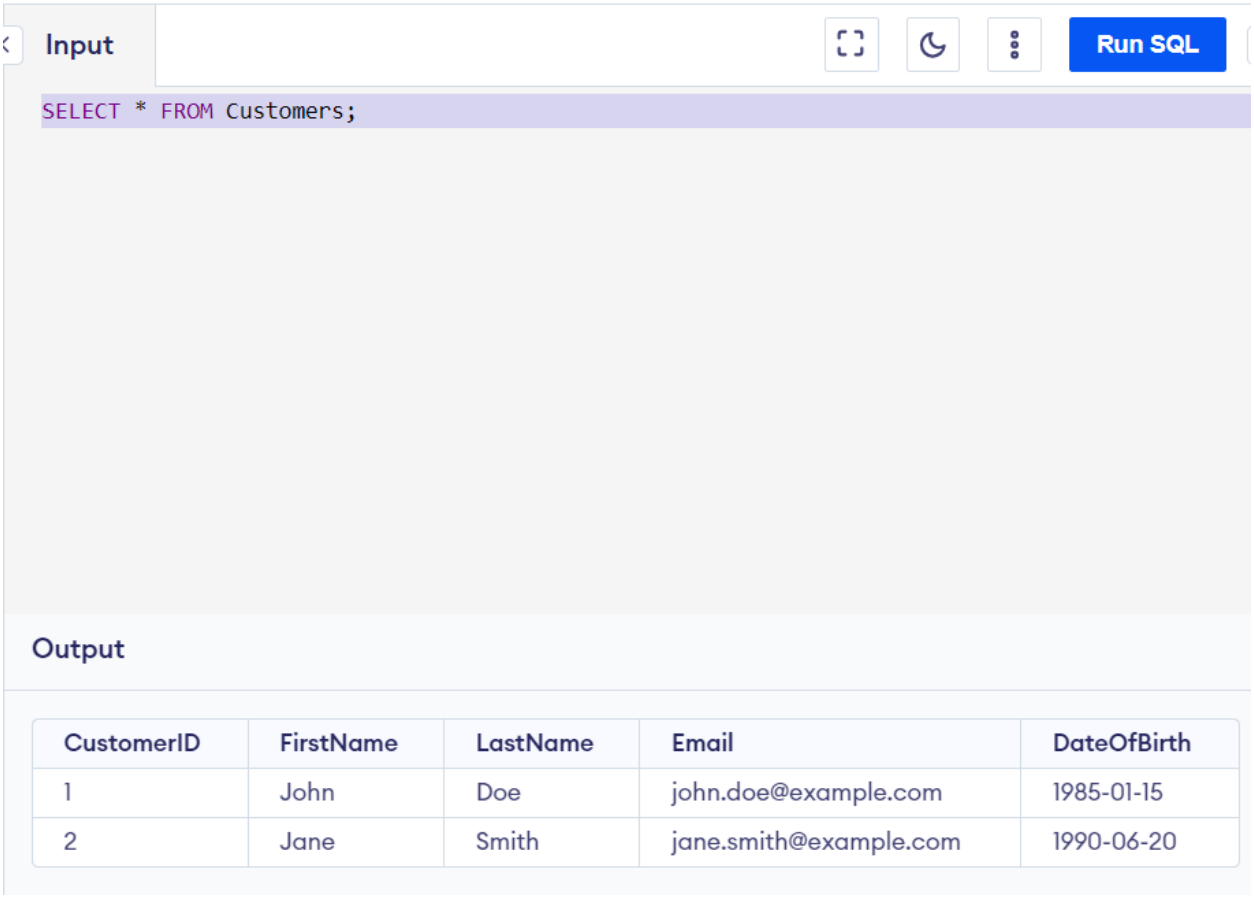
SELECT strftime('%Y-%m', OrderDate) AS Month, COUNT(*) AS TotalOrders, SUM(Products.Price * OrderItems.Quantity) AS TotalSales FROM Orders JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID JOIN Products ON OrderItems.ProductID = Products.ProductID WHERE strftime('%Y', OrderDate) = '2023' GROUP BY strftime('%Y-%m', OrderDate);

8.Find customers who have spent more than $1000.

SELECT Customers.CustomerID, Customers.FirstName, Customers.LastName, SUM(Products.Price * OrderItems.Quantity) AS TotalSpent FROM Customers JOIN Orders ON Customers.CustomerID = Orders.CustomerID JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID JOIN Products ON

OrderItems.ProductID = Products.ProductID GROUP BY Customers.CustomerID, Customers.FirstName, Customers.LastName HAVING TotalSpent > 1000;

OUTPUTS FOR SQL QUERIES:



CustomerID	FirstName	LastName	Email	DateOfBirth
1	John	Doe	john.doe@example.com	1985-01-15
2	Jane	Smith	jane.smith@example.com	1990-06-20

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT * FROM Orders
WHERE OrderDate BETWEEN '2023-01-01' AND '2023-01-31';
```

Output

OrderID	CustomerID	OrderDate
1	1	2023-01-10
2	2	2023-01-12

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT Orders.OrderID, Customers.FirstName, Customers.LastName, Customers.Email,
Orders.OrderDate
FROM Orders
JOIN Customers ON Orders.CustomerID = Customers.CustomerID;
```

Output

OrderID	FirstName	LastName	Email	OrderDate
1	John	Doe	john.doe@example.com	2023-01-10
2	Jane	Smith	jane.smith@example.com	2023-01-12

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT Products.ProductName, OrderItems.Quantity
FROM OrderItems
JOIN Products ON OrderItems.ProductID = Products.ProductID
WHERE OrderItems.OrderID = 1;
```

Output

ProductName	Quantity
Laptop	1
Headphones	2

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT Customers.CustomerID, Customers.FirstName, Customers.LastName, SUM(Products.Price *
OrderItems.Quantity) AS TotalSpent
FROM Customers
JOIN Orders ON Customers.CustomerID = Orders.CustomerID
JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID
JOIN Products ON OrderItems.ProductID = Products.ProductID
GROUP BY Customers.CustomerID, Customers.FirstName, Customers.LastName;
```

Output

CustomerID	FirstName	LastName	TotalSpent
1	John	Doe	1200
2	Jane	Smith	700

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT Products.ProductID, Products.ProductName, SUM(OrderItems.Quantity) AS TotalOrdered
FROM OrderItems
JOIN Products ON OrderItems.ProductID = Products.ProductID
GROUP BY Products.ProductID, Products.ProductName
ORDER BY TotalOrdered DESC
LIMIT 1;
```

Output

ProductID	ProductName	TotalOrdered
3	Headphones	3

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT strftime('%Y-%m', OrderDate) AS Month,
       COUNT(*) AS TotalOrders,
       SUM(Products.Price * OrderItems.Quantity) AS TotalSales
FROM Orders
JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID
JOIN Products ON OrderItems.ProductID = Products.ProductID
WHERE strftime('%Y', OrderDate) = '2023'
GROUP BY strftime('%Y-%m', OrderDate);
```

Output

Month	TotalOrders	TotalSales
2023-01	4	1900

210030140
D BALAJI REDDY

Input

Run SQL

```
SELECT Customers.CustomerID, Customers.FirstName, Customers.LastName, SUM(Products.Price *
OrderItems.Quantity) AS TotalSpent
FROM Customers
JOIN Orders ON Customers.CustomerID = Orders.CustomerID
JOIN OrderItems ON Orders.OrderID = OrderItems.OrderID
JOIN Products ON OrderItems.ProductID = Products.ProductID
GROUP BY Customers.CustomerID, Customers.FirstName, Customers.LastName
HAVING TotalSpent > 1000;
```

Output

CustomerID	FirstName	LastName	TotalSpent
1	John	Doe	1200