Case Study Mysql Online Retail Store

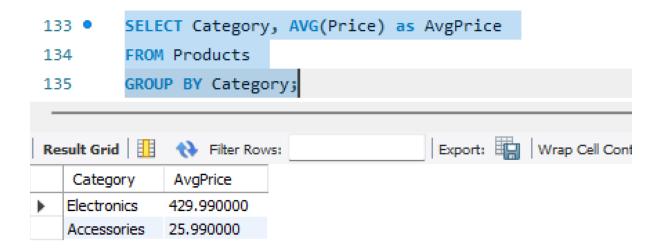
- 1. Write SQL statements to create the tables as described in the database schema.
- 2. Insert Data
- 3. Retrieve all customers who registered in 2023.

```
CREATE DATABASE OnlineRetailStore;
    2 •
             USE OnlineRetailStore;
    4 • ○ CREATE TABLE Customers (
    5
                    CustomerID INT PRIMARY KEY,
                    FirstName VARCHAR(255),
    6
    7
                    LastName VARCHAR(255),
                    Email VARCHAR(255),
    8
                    Phone VARCHAR(255),
    9
                    RegistrationDate DATE
  10
  11
              );
      INSERT INTO Customers VALUES
      (1, 'John', 'Doe', 'john.doe@example.com', '123-456-7890', '2023-01-15'),
13
      (2, 'Jane', 'Smith', 'jane.smith@example.com', '234-567-8901', '2023-02-20'),
14
15
      (3, 'Alice', 'Johnson', 'alice.johnson@example.com', '345-678-9012', '2023-03-10'),
      (4, 'Bob', 'Brown', 'bob.brown@example.com', '456-789-0123', '2023-04-05'),
16
      (5, 'Charlie', 'Davis', 'charlie.davis@example.com', '567-890-1234', '2023-05-12'),
17
      (6, 'David', 'Wilson', 'david.wilson@example.com', '678-901-2345', '2023-06-15'),
18
19
      (7, 'Emma', 'Thomas', 'emma.thomas@example.com', '789-012-3456', '2023-07-01'),
      (8, 'Fiona', 'Garcia', 'fiona.garcia@example.com', '890-123-4567', '2023-07-10'),
20
      (9, 'George', 'Martinez', 'george.martinez@example.com', '901-234-5678', '2023-07-20'),
22
      (10, 'Hannah', 'Rodriguez', 'hannah.rodriguez@example.com', '012-345-6789', '2023-07-25');
```

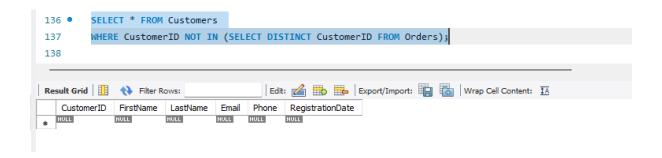
```
INSERT INTO Products VALUES
32 •
       (1, 'Laptop', 'Electronics', 999.99, 50),
33
       (2, 'Smartphone', 'Electronics', 499.99, 100),
34
       (3, 'Tablet', 'Electronics', 299.99, 75),
35
       (4, 'Headphones', 'Accessories', 49.99, 200),
36
       (5, 'Charger', 'Accessories', 19.99, 300),
37
       (6, 'Keyboard', 'Accessories', 29.99, 150),
38
       (7, 'Mouse', 'Accessories', 19.99, 250),
39
       (8, 'Monitor', 'Electronics', 199.99, 30),
40
       (9, 'Printer', 'Electronics', 149.99, 20),
41
       (10, 'USB Cable', 'Accessories', 9.99, 400);
42
44 • ⊖ CREATE TABLE Orders (
45
         OrderID INT PRIMARY KEY,
         OrderDate DATE,
46
47
         CustomerID INT,
         TotalAmount DECIMAL(10, 2),
48
49
         FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)
50
     · );
51
 52 •
         INSERT INTO Orders VALUES
 53
         (1, '2023-06-01', 1, 1049.98),
         (2, '2023-06-05', 2, 549.98),
 54
         (3, '2023-06-10', 3, 999.99),
 55
         (4, '2023-06-15', 4, 69.98),
 56
 57
         (5, '2023-06-20', 5, 519.98),
         (6, '2023-06-25', 6, 229.98),
 58
         (7, '2023-07-02', 7, 1199.97),
 59
         (8, '2023-07-12', 8, 49.98),
 60
         (9, '2023-07-18', 9, 349.98),
 61
         (10, '2023-07-22', 10, 39.98);
 62
 63
```

```
64 • ⊖ CREATE TABLE OrderDetails (
65
               OrderDetailID INT PRIMARY KEY,
               OrderID INT,
66
               ProductID INT,
67
               Quantity INT,
68
               Price DECIMAL(10, 2),
69
               FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),
70
               FOREIGN KEY (ProductID) REFERENCES Products(ProductID)
71
72
          );
    INSERT INTO OrderDetails VALUES
    (1, 1, 1, 1, 999.99), (2, 1, 4, 1, 49.99), (3, 2, 2, 1, 499.99), (4, 2, 5, 1, 49.99), (5, 3, 1, 1, 999.99),
    (6, 4, 4, 1, 49.99), (7, 4, 5, 1, 19.99), (8, 5, 2, 1, 499.99), (9, 5, 5, 1, 19.99), (10, 6, 3, 1, 199.99),
    (11, 6, 5, 1, 29.99), (12, 7, 1, 1, 999.99), (13, 7, 3, 1, 199.99), (14, 8, 7, 1, 19.99), (15, 8, 8, 1, 29.99),
    (16, 9, 4, 2, 149.99), (17, 9, 9, 1, 49.99), (18, 10, 10, 4, 9.99);
) • SELECT * FROM Customers;
SELECT * FROM Products;
   SELECT * FROM Orders;
    SELECT * FROM OrderDetails;
3. Retrieve all customers who registered in 2023:
 129 •
           SELECT FirstName, LastName, SUM(TotalAmount) as TotalSpent
 130
           FROM Customers
           JOIN Orders ON Customers.CustomerID = Orders.CustomerID
 131
           GROUP BY Customers.CustomerID;
                                              Export: Wrap Cell Content: 1A
 FirstName LastName TotalSpent
    John
              Doe
                         1049.98
    Jane
              Smith
                         549.98
    Alice
              Johnson
                         999.99
    Bob
                         69.98
              Brown
    Charlie
              Davis
                         519.98
              Wilson
                        229.98
    David
    Emma
              Thomas
                         1199.97
    Fiona
              Garcia
                         49.98
    George
              Martinez
                         349.98
    Hannah
              Rodriguez 39.98
```

4. List all products in the 'Electronics' category.



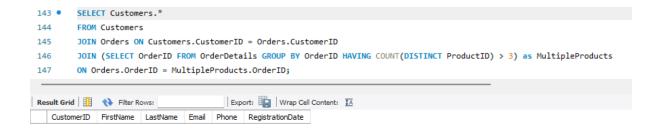
5. Find the total number of orders placed by each customer.



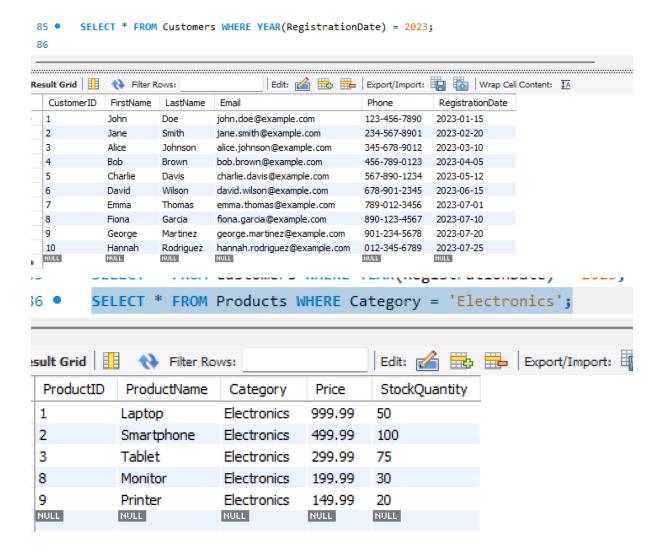
6. Calculate the total sales amount for each product.

```
SELECT ProductName, SUM(Products.Price * Quantity) as TotalSales
138 •
139
        FROM OrderDetails
140
        JOIN Products ON OrderDetails.ProductID = Products.ProductID
        GROUP BY ProductName
141
        ORDER BY TotalSales DESC LIMIT 3;
142
143
144
                                       Export: Wrap Cell Content: 🖽 Fetch rows:
ProductName
              TotalSales
  Laptop
              2999.97
  Smartphone
             999.98
  Tablet
             599.98
```

7. Retrieve the details of all orders, including the customer name and total amount.

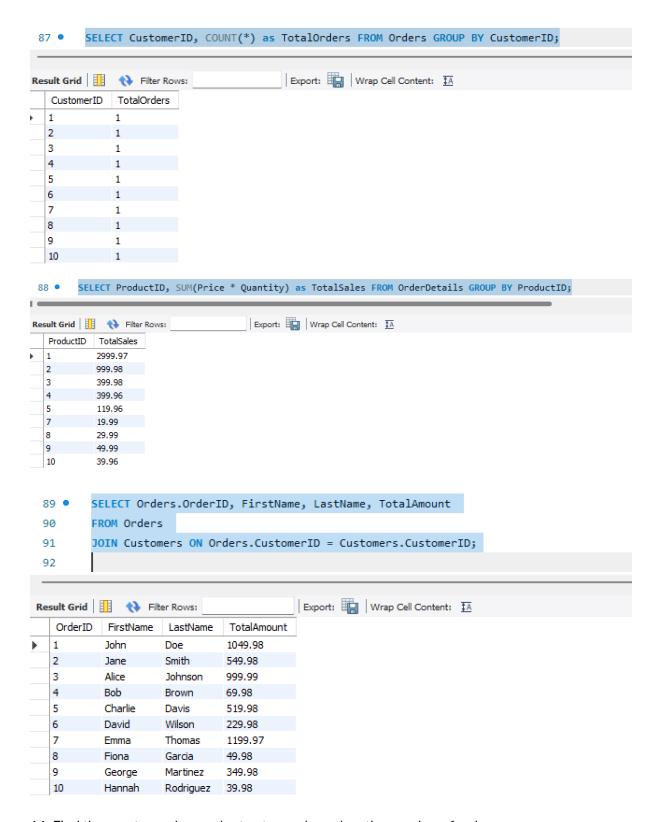


- 8. List all products that have been ordered along with the quantity ordered for each.
- 9. Find the order details for orders placed by 'John Doe'.
- 10. Find customers who have placed an order totaling more than \$500.



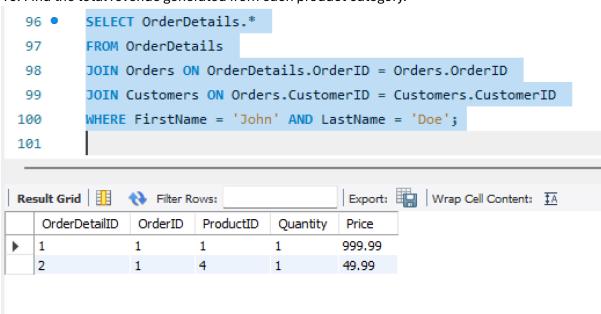
11. List the products that have never been ordered.

- 12. Retrieve the order history for a specific customer, including order date, product names, and quantities. (Assume customer ID is 2)
- 13. Calculate the average order value for each customer.



14. Find the most popular product category based on the number of orders.

- 15. List all customers who have ordered more than one product in a single order.
- 16. Find the total revenue generated from each product category.



```
SELECT ProductName, SUM(Quantity) as TotalQuantity
 93
           FROM OrderDetails
          JOIN Products ON OrderDetails.ProductID = Products.ProductID
 94
 95
           GROUP BY ProductName;
Result Grid
                 Filter Rows:
                                                    Export: Wrap Cell Content: $\frac{1}{4}$
   ProductName
                  TotalQuantity
   Laptop
                  3
   Headphones
                  4
   Smartphone
                  2
                  4
   Charger
   Tablet
                  2
   Mouse
                  1
   Monitor
                  1
   Printer
                  1
   USB Cable
                  4
 101 •
          SELECT Customers.*
 102
          FROM Customers
          JOIN Orders ON Customers.CustomerID = Orders.CustomerID
 103
          WHERE TotalAmount > 500;
 104
 105
 Result Grid
                                            Export: Wrap Cell Content: 1A
              Filter Rows:
    CustomerID
               FirstName
                         LastName
                                                           Phone
                                                                        RegistrationDate
               John
                                  john.doe@example.com
                                                          123-456-7890
                                                                       2023-01-15
   2
               Jane
                         Smith
                                  jane.smith@example.com
                                                          234-567-8901 2023-02-20
    3
               Alice
                         Johnson
                                  alice.johnson@example.com
                                                          345-678-9012
                                                                       2023-03-10
   5
                         Davis
                                  charlie.davis@example.com
                                                          567-890-1234 2023-05-12
               Emma
                         Thomas
                                  emma.thomas@example.com
                                                          789-012-3456
                                                                       2023-07-01
```

17. Retrieve the list of customers along with the total amount they have spent.

- 18. Find the average price of products in each category.
- 19. Find all customers who have not placed any orders.
- 20. List the top 3 products with the highest total sales amount.
- 21. Find customers who have placed orders for more than 3 different products

