SQL-Case Study: Online Retail Store

1. Write SQL statements to create the tables as described in the database schema.

```
16 • ⊖ create table product_table(product_id int primary key,
                                                                   17
                                                                           product_name varchar (100),
7 • ⊖ create table customers_table(customer_id int primary key,
                                                                          category varchar (100).
                                                                   18
      first name varchar (100).
      last name varchar (100),
                                                                         price decimal (5,2),
      email varchar (100),
                                                                   20
                                                                          stock_qty int);
     phone varchar (100),
                                                                    21 • select * from product_table;
      registration_date date );
13 • select * from customers_table;
                                                                    34 • ⊖ create table order_details_table(Order_detail_ID INT Primary Key,
                                                                    35
                                                                            order id INT.
26 • ⊖ create table orders_table(order_id int primary key,
                                                                            foreign key(order_id) references orders_table(order_id),
      order date date,
                                                                            product_id int,
28
      customer id int,
                                                                     37
      foreign key(customer_id) references customers_table(customer_id), 38
                                                                            foreign key(product_id) references product_table(product_id),
     total_amt decimal(5,2));
30
                                                                            quantity int,
31 • select * from orders table;
                                                                     40
                                                                            price decimal(5,2));
```

2. Insert Data

```
insert into customers_table values
                                                                                                    55 • insert into product table values
      (1,'John', 'Doe', 'john.doe@example.com', '123-456-7890', '2023-01-15'),
                                                                                                           (101, 'Laptop', 'Electronics', 999.99, 50),
      (2, 'Jane', 'Smith', 'jane.smith@example.com', '234-567-8901', '2023-02-20'),
                                                                                                   57 (102, 'Smartphone', 'Electronics', 499.99, 100),
                                                                                                  58 (103, 'Tablet', 'Electronics', 299.99, 75),
      (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'),
                                                                                                    59
                                                                                                           (104, 'Headphones', 'Accessories', 49.99, 200),
      (5, 'Charlie', 'Davis', 'charlie.davis@example.com', '567-890-1234', '2023-05-12'),
                                                                                                  60 (105, 'Charger', 'Accessories', 19.99, 300),
      (6,'David', 'Wilson', 'david.wilson@example.com', '678-901-2345', '2023-06-15'),
                                                                                                   61 (106, 'Keyboard', 'Accessories', 29.99, 150),
      (7,'Emma', 'Thomas', 'emma.thomas@example.com', '789-012-3456', '2023-07-01'),
                                                                                                           (107, 'Mouse', 'Accessories', 19.99, 250),
      (8, 'Fiona', 'Garcia', 'fiona.garcia@example.com', '890-123-4567', '2023-07-10'),
                                                                                                   63 (108, 'Monitor', 'Electronics', 199.99, 30),
      (9, 'George', 'Martinez', 'george.martinez@example.com', '901-234-5678', '2023-07-20'),
      (10, 'Hannah', 'Rodriguez', 'hannah.rodriguez@example.com', '012-345-6789', '2023-07-25');
                                                                                                   64 (109, 'Printer', 'Electronics', 149.99, 20),
                                                                                                           (110, 'USB Cable', 'Accessories', 9.99, 400);
67 • insert into orders_table values
                                                                                                        79 • insert into order_details_table values
       (201, '2023-06-01', 1, 104.98),
68
                                                                                                               (301, 201, 101, 1, 999.99),
     (202, '2023-06-05', 2, 549.98),
                                                                                                               (302, 204, 103, 2, 49.99),
                                                                                                               (303, 202, 101, 1, 499.99),
70
     (203,'2023-06-10',3,999.99),
                                                                                                               (304, 205, 101, 10, 49.99),
       (204, '2023-06-15', 4, 69.98),
71
                                                                                                              (305, 201, 105, 5, 999.99),
       (205, '2023-06-20', 5, 519.98),
                                                                                                               (306, 204, 101, 7, 49.99),
                                                                                                               (307, 205, 110, 4, 19.99),
       (206, '2023-06-25', 6, 229.98),
                                                                                                               (308, 202, 109, 12, 499.99),
      (207, '2023-07-02', 7, 119.97),
                                                                                                               (309, 205, 105, 1, 19.99),
75
     (208, '2023-07-12',8,49.98),
                                                                                                               (310, 203, 103, 19, 199.99),
       (209, '2023-07-18', 9, 349.98),
76
                                                                                                               (311, 205, 110, 12, 29.99),
       (210, '2023-07-22', 10, 39.98);
                                                                                                               (312, 201, 106, 25, 999.99),
                                                                                                               (313, 203, 107, 6, 199.99),
                                                                                                               (314, 201, NULL, 8, 49.99),
                                                                                                               (315, 207, 108, 20, 19.99),
                                                                                                               (316, 208, 102, 35, 29.99),
                                                                                                               (317, 204, 110, 40, 149.99),
                                                                                                               (318, 209, 101, 15, 49.99),
                                                                                                               (319, 210, 104, 30, 9.99);
```

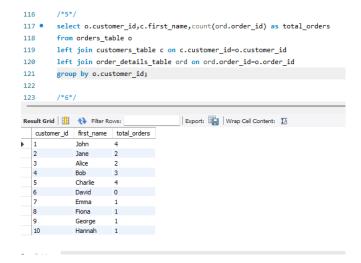
3. Retrieve all customers who registered in 2023.



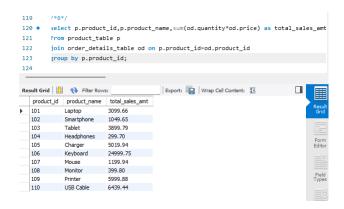
4. List all products in the 'Electronics' category.



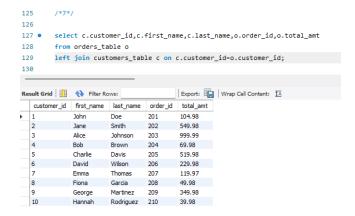
5. Find the total number of orders placed by each customer.



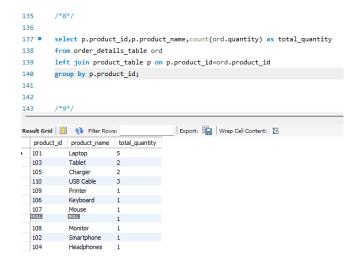
6. Calculate the total sales amount for each product



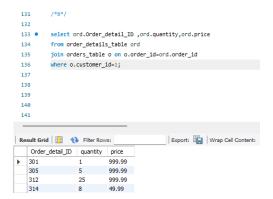
7. Retrieve the details of all orders, including the customer name and total amount.



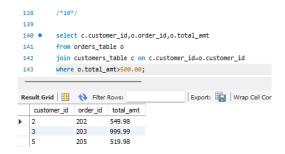
8. List all products that have been ordered along with the quantity ordered for each.



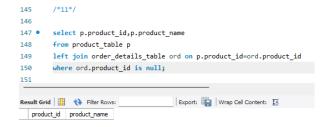
9. Find the order details for orders placed by 'John Doe'.



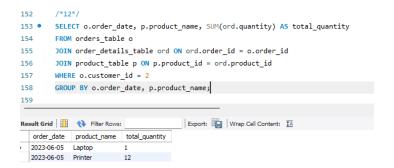
10. Find customers who have placed an order totaling more than \$500.



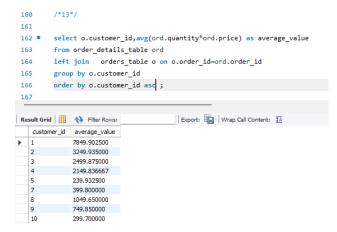
11. List the products that have never been ordered.



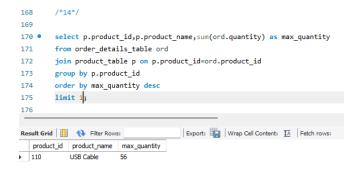
12. Retrieve the order history for a specific customer, including order date, product names, and quantities. (Assume customer ID is 2)



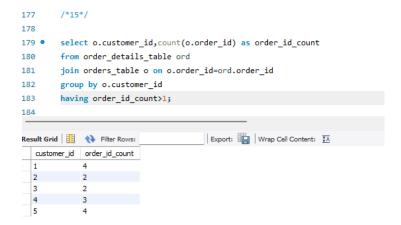
13. Calculate the average order value for each customer.



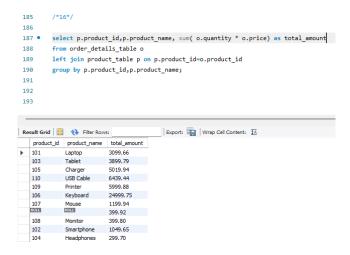
14. Find the most popular product category based on the number of orders.



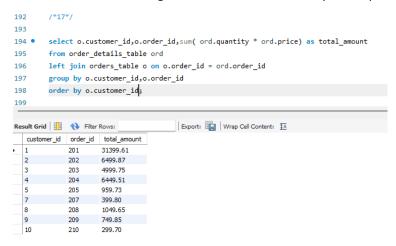
15. List all customers who have ordered more than one product in a single order.



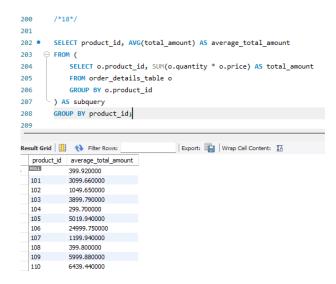
16. Find the total revenue generated from each product category.



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



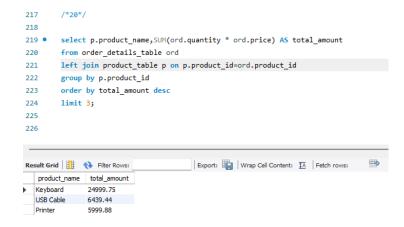
18. Find the average price of products in each category.



19. Find all customers who have not placed any orders.

```
222
       /*19*/
223
224 • select o.customer_id,o.order_id,count(ord.order_id) as total_orders
      from order_details_table ord
      left join orders_table o on o.order_id=ord.order_id
226
227
       group by o.order_id,o.customer_id
228
      having total_orders=0;
229
230
                                  Export: Wrap Cell Content: 🖽
customer_id order_id total_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.

```
226
227
228 •
      select o.customer_id,o.order_id,count(ord.order_id) as total_orders
229
       from order_details_table ord
      left join orders_table o on o.order_id=ord.order_id
230
231
      group by o.order_id
232
       having total_orders > 3;
233
Export: Wrap Cell Content: ‡A
 customer_id order_id total_orders
           201
 1
    201 4
```