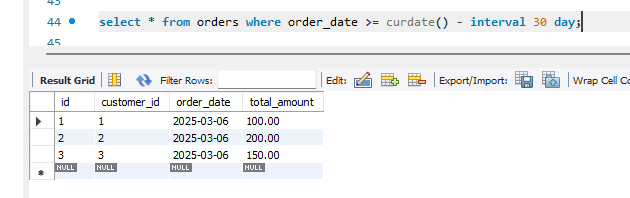
Qs 1: Retrieve all customers who have placed an order in the last 30 days.

* **select \* from orders where order\_date >= curdate() - interval 30 day;**



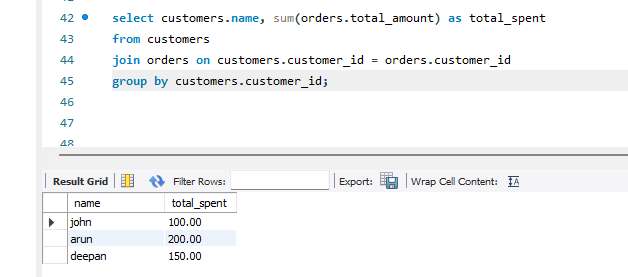
Qs 2: Get the total amount of all orders placed by each customer.

* **select customers.name, sum(orders.total\_amount) as total\_spent**

**from customers**

**join orders on customers.customer\_id = orders.customer\_id**

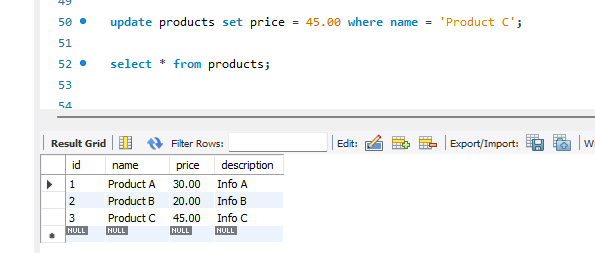
**group by customers.customer\_id;**



Qs 3: Update the price of Product C to 45.00.

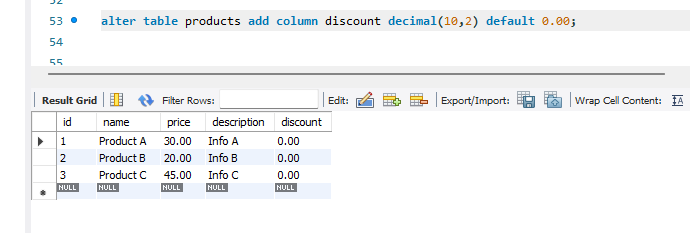
* **update products set price = 45.00 where name = 'Product C';**

**select \* from products;**



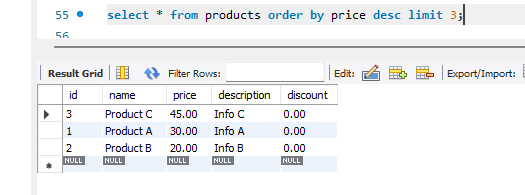
Qs 4: Add a new column discount to the products table.

* **alter table products add column discount decimal(10,2) default 0.00;**



Qs 5: Retrieve the top 3 products with the highest price.

* **select \* from products order by price desc limit 3;**



Qs 7: Get the names of customers who have ordered Product A.

* **select distinct customers.name**

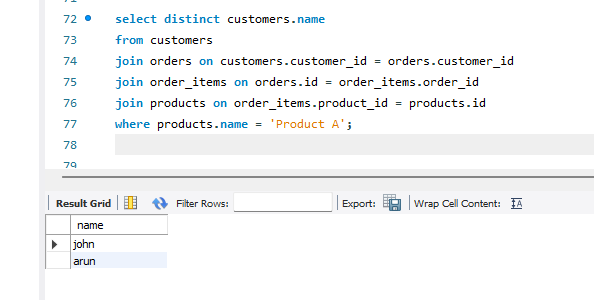
**from customers**

**join orders on customers.customer\_id = orders.customer\_id**

**join order\_items on orders.id = order\_items.order\_id**

**join products on order\_items.product\_id = products.id**

**where products.name = 'Product A';**

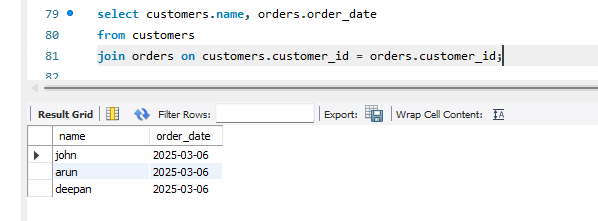


Qs 8: Join the orders and customers tables to retrieve the customer's name and order date for each order.

* **select customers.name, orders.order\_date**

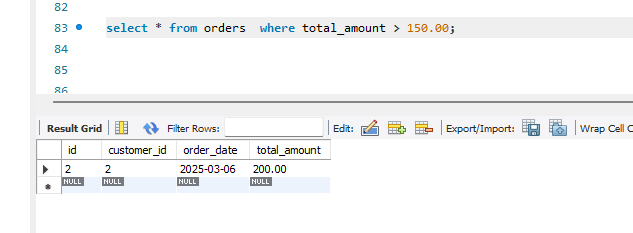
**from customers**

**join orders on customers.customer\_id = orders.customer\_id;**



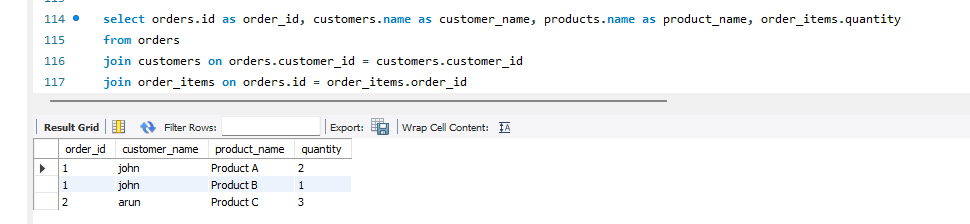
Qs 9: Retrieve the orders with a total amount greater than 150.00.

* **select \* from orders where total\_amount > 150.00;**



Qs 10: Normalize the database by creating a separate table for order items and updating the orders table to reference the order\_items table.

* **select \* from orders where total\_amount > 150.00;**



Qs 11: Retrieve the average total of all orders.

* **select avg(total\_amount) as average from orders;**

