**Module 9 Assignment – DB**

**TASK 1**

**Conceptual Model:**

* Customers (1) → Orders (N): Each order belongs to one customer.
* Orders (1) → Order\_Items (N): Each order has multiple order items.
* Products (1) → Order\_Items (N): Each product can appear in multiple order items.
* Order\_Items connects orders and products to handle many-to-many.

-------------------------------------

**Customers**

-------------------------------------

customer\_id (PK)

name

email

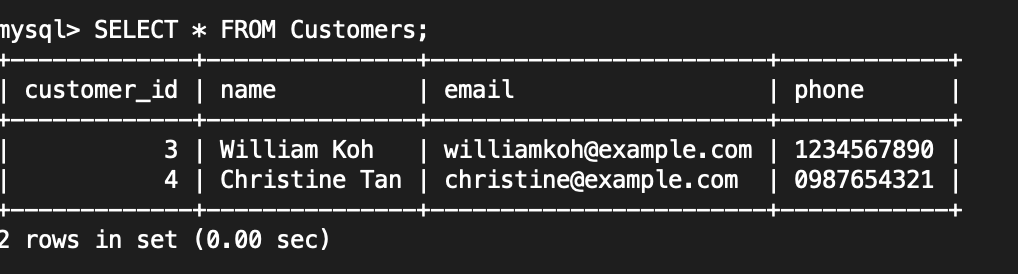
phone

**COMAND** –

INSERT INTO Customers (name, email, phone) VALUES

('William Koh', 'williamkoh@example.com', '123-456-7890'),

('Christine Tan', 'christine@example.com', '098-765-4321');



--------------------------------------

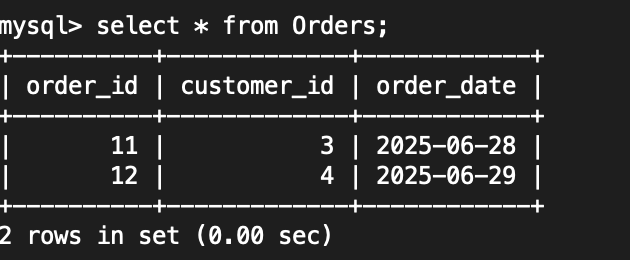
**Orders**

-------------------------------------

order\_id (PK)

customer\_id (FK)

order\_date



**COMMAND** –

INSERT INTO Orders (customer\_id, order\_date) VALUES

(3, '2025-06-28'),

(4, '2025-06-29');

-------------------------------------

**Products**

-------------------------------------

product\_id (PK)

name

price

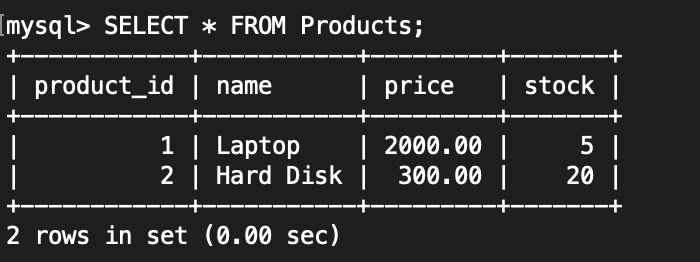
stock

**COMMAND**:

INSERT INTO Products (name, price, stock) VALUES

('Laptop', 2000.00, 5),

('Hard Disk', 300.00, 20);



-------------------------------------

**Order\_Items**

-------------------------------------

order\_item\_id (PK)

order\_id (FK)

product\_id (FK)

quantity

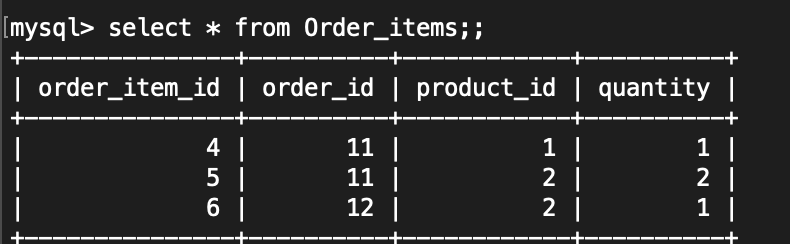
**COMMAND**:

INSERT INTO Order\_Items (order\_id, product\_id, quantity) VALUES

(11, 1, 1), -- For order 11, 1 unit of product 1

(11, 2, 2), -- For order 11, 2 units of product 2

(12, 2, 1); -- For order 12, 1 unit of product 2



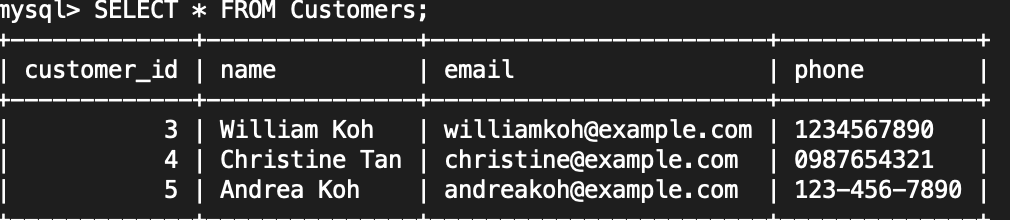
**TASK 2**

a) Insert a new customer record

INSERT INTO Customers (name, email, phone)

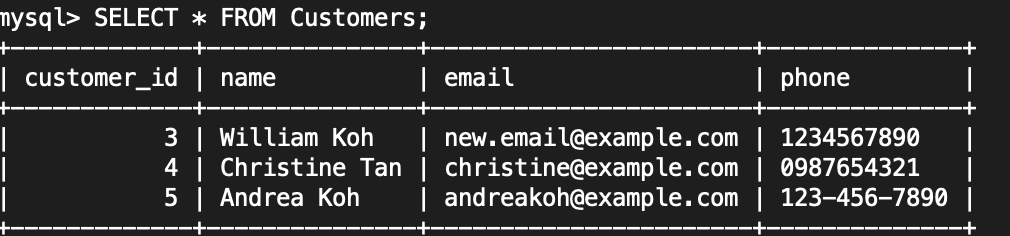
VALUES ('Andrea Koh', 'andreakoh@example.com', '123-456-7890');

Note -(customer\_id because it is AUTO\_INCREMENT)



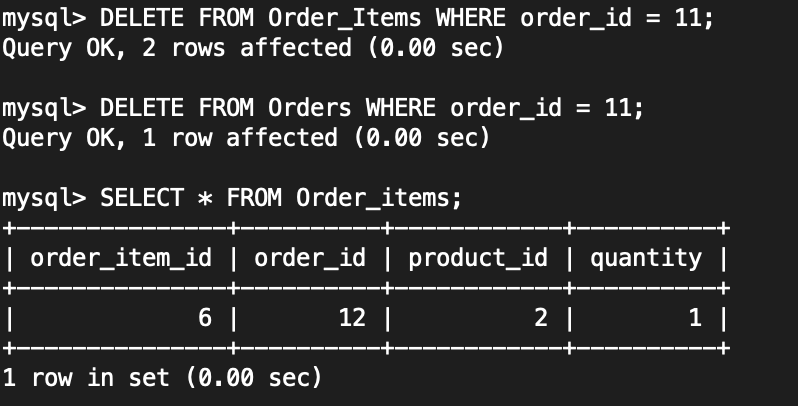
b) Update a customer's email address based on their customer\_id

Assuming update the customer with customer\_id = 3:



### c) Delete a specific record from the Orders table based on order\_id

Assuming you want to delete the order with order\_id = 11:



Note: If there's a related row in the Order\_Items table that refers to order\_id = 11.

MySQL won't let you delete the parent (Orders) unless the child (Order\_Items) is removed first (or updated).

**TASK 3**

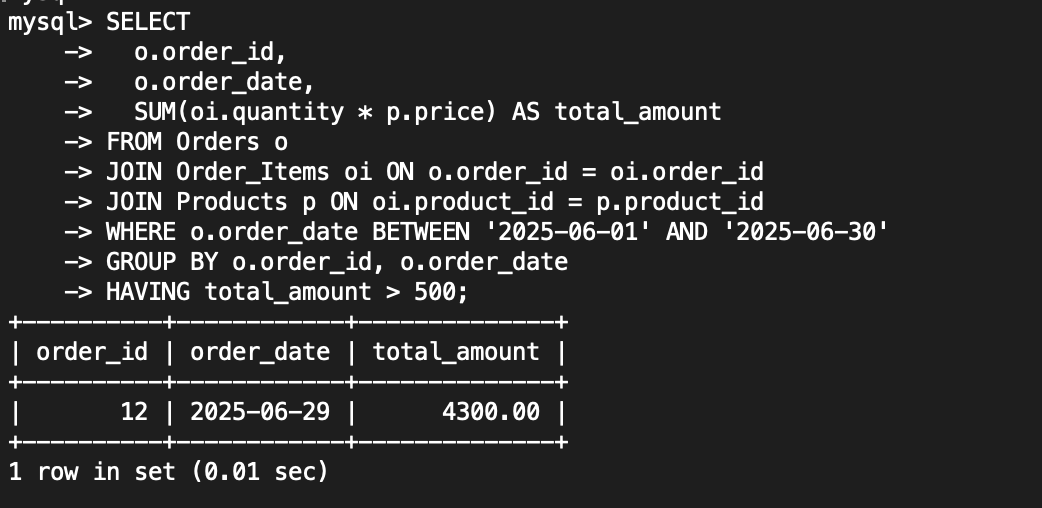
**3. Advanced SELECT Queries: Filtered Data Retrieval (5 Marks)**

Write a SELECT query to fetch all orders from the Orders table that meet the following criteria:

a) Orders were placed in January 2024.

b) The order total exceeds $500.

c) Your query should only include columns for OrderID, OrderDate, and TotalAmount in the result.



 order\_id: the ID of the qualifying order.

 order\_date: the date the order was placed (in January 2024).

 total\_amount: the sum of all (quantity × product price) for that order.

**TASK 4**

**4. SQL Joins for Data Aggregation (5 Marks)**

Retrieve Information Using Joins

a) Using the tables Products, Orders, and OrderDetails, write an SQL query.

b) The query should retrieve information on products ordered in June 2024.

c) The result should display:

ProductName: The name of the product.

QuantityOrdered: The quantity of each product in the order.

OrderDate: The date the order was placed.

