

SQL Intern Task-1

Name:Swetha R

Phone:6380 459 779

Mail:rswetha2807@gmail.com

1.Domain:

E-commerce

This system allows customers to browse products, place orders, and track shipments.

2. Entities and Relationships:

Entities:

1. Customer
2. Product
3. Category
4. Order
5. OrderItem
6. Payment
7. Shipment

Relationships:

- A Customer can place many Orders.
- An Order can contain many OrderItems.
- Each OrderItem refers to one Product.
- Each Product belongs to one Category.
- An Order has one Payment.
- An Order can have one Shipment.

SQL Intern Task-1

3.CREATE TABLE statements with Primary Key and Foreign Key

```
CREATE TABLE Customer (
```

```
    customer_id INT PRIMARY KEY,
```

```
    name VARCHAR(100),
```

```
    email VARCHAR(100) UNIQUE,
```

```
    phone VARCHAR(15),
```

```
    address TEXT
```

```
);
```

```
CREATE TABLE Category (
```

```
    category_id INT PRIMARY KEY,
```

```
    category_name VARCHAR(100)
```

```
);
```

```
CREATE TABLE Product (
```

```
    product_id INT PRIMARY KEY,
```

```
    name VARCHAR(100),
```

```
    description TEXT,
```

```
    price DECIMAL(10,2),
```

```
    category_id INT,
```

```
    FOREIGN KEY (category_id) REFERENCES Category(category_id)
```

```
);
```

SQL Intern Task-1

```
CREATE TABLE Orders (  
    order_id INT PRIMARY KEY,  
    customer_id INT,  
    order_date DATE,  
    total_amount DECIMAL(10,2),  
    FOREIGN KEY (customer_id) REFERENCES Customer(customer_id)  
);
```

```
CREATE TABLE OrderItem (  
    order_item_id INT PRIMARY KEY,  
    order_id INT,  
    product_id INT,  
    quantity INT,  
    price DECIMAL(10,2),  
    FOREIGN KEY (order_id) REFERENCES Orders(order_id),  
    FOREIGN KEY (product_id) REFERENCES Product(product_id)  
);
```

```
CREATE TABLE Payment (  
    payment_id INT PRIMARY KEY,  
    order_id INT,  
    payment_date DATE,  
    amount DECIMAL(10,2),  
    payment_method VARCHAR(50),  
    FOREIGN KEY (order_id) REFERENCES Orders(order_id)
```

SQL Intern Task-1

);

CREATE TABLE Shipment (

shipment_id INT PRIMARY KEY,

order_id INT,

shipment_date DATE,

delivery_date DATE,

status VARCHAR(50),

FOREIGN KEY (order_id) REFERENCES Orders(order_id)

);

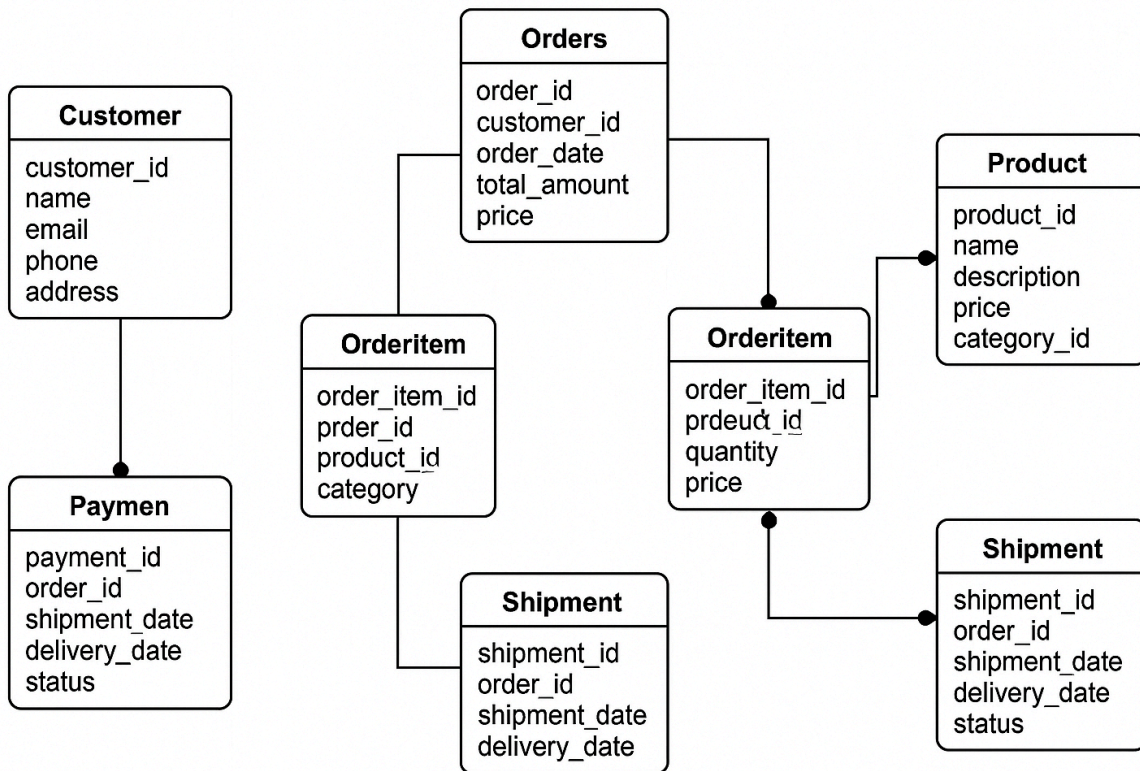
SQL Intern Task-1

4. Primary and Foreign Keys

Table Name	Primary Key	Foreign Keys
Customer	<code>customer_id</code> —	
Category	<code>category_id</code> —	
Product	<code>product_id</code>	<code>category_id</code> → Category
Orders	<code>order_id</code>	<code>customer_id</code> → Customer
OrderItem	<code>order_item_id</code>	<code>order_id</code> → Orders,
Payment	<code>payment_id</code>	<code>order_id</code> → Orders
Shipment	<code>shipment_id</code>	<code>order_id</code> → Orders

ER:

SQL Intern Task-1



EXAMPLE SCHEMA:

SQL Intern Task-1

✓ 1. CUSTOMER Table Example

```
INSERT INTO Customer (customer_id, name, email, phone, address)
VALUES (1, 'Arjun Kumar', 'arjun.k@example.com', '9876543210', '123, MG Road, Chennai');
```

✓ 2. CATEGORY Table Example

```
INSERT INTO Category (category_id, category_name)
VALUES (1, 'Electronics');
```

✓ 3. PRODUCT Table Example

```
INSERT INTO Product (product_id, name, description, price, category_id)
VALUES (101, 'Bluetooth Headphones', 'Noise-canceling over-ear headphones', 2999.99, 1);
```

✓ 4. ORDERS Table Example

```
INSERT INTO Orders (order_id, customer_id, order_date, total_amount)
VALUES (5001, 1, '2025-06-23', 2999.99);
```

✓ 5. ORDERITEM Table Example

```
INSERT INTO OrderItem (order_item_id, order_id, product_id, quantity, price)
VALUES (9001, 5001, 101, 1, 2999.99);
```

✓ 6. PAYMENT Table Example

```
INSERT INTO Payment (payment_id, order_id, payment_date, amount, payment_method)
VALUES (7001, 5001, '2025-06-23', 2999.99, 'UPI');
```

✓ 7. SHIPMENT Table Example

```
INSERT INTO Shipment (shipment_id, order_id, shipment_date, delivery_date, status)
VALUES (8001, 5001, '2025-06-24', '2025-06-26', 'In Transit');
```