MD

Elevate Labs SQL Developer Internship

Task 1 - Database Setup and Schema Design.

- **Domain: E-commerce :**
- **Entities -**
    1. Customer
    2. Address
    3. Category
    4. Product
    5. Order
    6. OrderItem
    7. Payment
    8. Shipment
    9. Cart

- **Realationships –**
    1. A Customer can have many Orders (1:N)
    2. A **Customer** can have many **Addresses** (1:N) — addresses for shipping, billing
    3. A **Category** has many **Products** (1:N)
    4. An **Order** has many **OrderItems** (1:N)
    5. Each **OrderItem** refers to one **Product** (N:1)
    6. An **Order** has one **Payment**
    7. An **Order** has one **Shipment** (1:1)
    8. A **Customer** has one **Cart**
    9. A **Cart** has many **CartItems**
    10. **CartItem** refers to **Product**

MD

- **Create TABLE –**

1. Table: Customer

```
CREATE TABLE Customer (
   customer_id SERIAL PRIMARY KEY,
   first_name VARCHAR(100) NOT NULL,
   last_name VARCHAR(100) NOT NULL,
   email VARCHAR(255) NOT NULL UNIQUE,
   password_hash VARCHAR(255) NOT NULL,
   phone VARCHAR(20),
   created_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
   updated_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP
);
```

2. Table: Address

```
CREATE TABLE Address (
   address_id SERIAL PRIMARY KEY,
   customer_id INTEGER NOT NULL,
   street VARCHAR(255) NOT NULL,
   city VARCHAR(100) NOT NULL,
   state VARCHAR(100),
   postal_code VARCHAR(20),
   country VARCHAR(100) NOT NULL,
   FOREIGN KEY (customer_id) REFERENCES
Customer(customer_id)
);
```

3. Table: Category

```
CREATE TABLE Category (
   category_id SERIAL PRIMARY KEY,
   name VARCHAR(100) NOT NULL UNIQUE,
   description TEXT
);
```

4. Table: Product

```
CREATE TABLE Product (
    product_id SERIAL PRIMARY KEY,
    name VARCHAR(255) NOT NULL,
    description TEXT,
    price DECIMAL(10,2) NOT NULL,
    stock_quantity INTEGER NOT NULL DEFAULT 0,
    category_id INTEGER,
    created_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    updated_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    FOREIGN KEY (category_id) REFERENCES
Category(category_id)
);
```

5. Table: Order

```
CREATE TABLE "Order" (
    order_id SERIAL PRIMARY KEY,
    customer_id INTEGER NOT NULL,
    order_date TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    status VARCHAR(50) NOT NULL,
    total_amount DECIMAL(12,2) NOT NULL,
    shipping_address_id INTEGER,
    billing_address_id INTEGER,
    created_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    updated_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    FOREIGN KEY (customer_id) REFERENCES
Customer(customer_id),
```

```
    FOREIGN KEY (shipping_address_id) REFERENCES
Address(address_id),
    FOREIGN KEY (billing_address_id) REFERENCES
Address(address_id)
);
```

6. Table: OrderItem

```
CREATE TABLE OrderItem (
    order_item_id SERIAL PRIMARY KEY,
    order_id INTEGER NOT NULL,
    product_id INTEGER NOT NULL,
    quantity INTEGER NOT NULL,
    unit_price DECIMAL(10,2) NOT NULL,
    FOREIGN KEY (order_id) REFERENCES "Order"(order_id),
    FOREIGN KEY (product_id) REFERENCES Product(product_id)
);
```

7. Table: Payment

```
CREATE TABLE Payment (

    payment_id SERIAL PRIMARY KEY,

    order_id INTEGER NOT NULL UNIQUE,

    payment_date TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,

    amount DECIMAL(12,2) NOT NULL,

    payment_method VARCHAR(50) NOT NULL,

    status VARCHAR(50) NOT NULL,

    FOREIGN KEY (order_id) REFERENCES "Order"(order_id)

);
```

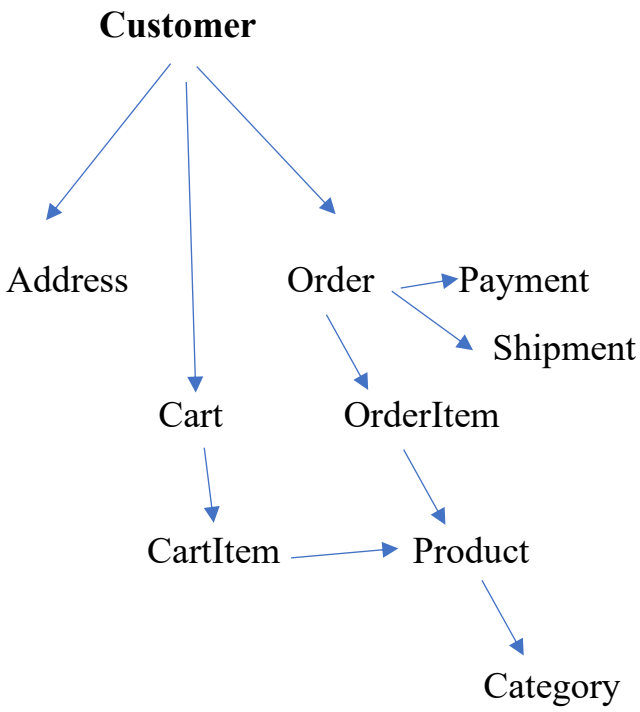
8. Table: Shipment

```
CREATE TABLE Shipment (
    shipment_id SERIAL PRIMARY KEY,
    order_id INTEGER NOT NULL UNIQUE,
    shipped_date TIMESTAMP,
    delivery_date TIMESTAMP,
    shipping_method VARCHAR(100),
    tracking_number VARCHAR(100),
    status VARCHAR(50),
    FOREIGN KEY (order_id) REFERENCES "Order"(order_id)
);
```

9. Table: Cart

```
CREATE TABLE Cart (
    cart_id SERIAL PRIMARY KEY,
    customer_id INTEGER NOT NULL UNIQUE,
    created_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    updated_at TIMESTAMP NOT NULL DEFAULT
CURRENT_TIMESTAMP,
    FOREIGN KEY (customer_id) REFERENCES
Customer(customer_id)
);
```

- **ER Diagram -**


Customer
Address
Order
Payment
Shipment
Cart
OrderItem
CartItem
Product
Category