

E-COMMERCE PROBLEM STATEMENT

You are tasked with building an e-commerce API using Django Rest Framework to manage customers, orders, and products.

The system should be able to handle multiple customers, each having multiple orders, and each order containing multiple products with corresponding quantities.

Database Structure:

- **Customer Model:**
 - Fields:
 - id (AutoField),
 - name (CharField)
 - contact_number (CharField)
 - email
- **Product Model:**
 - Fields:
 - id (AutoField),
 - name (CharField),
 - weight (DecimalField)
- **Order Model:**
 - Fields:
 - id (AutoField),
 - order_number (CharField),
 - Order Number needs be auto generated with prefix ORD and 00001 and so on.
 - Examples (ORD00001, ORD00002)
 - customer (ForeignKey to Customer),
 - order_date (DateField),
 - address (CharField),
- **Order Item Model:**
 - Fields:
 - id (AutoField),
 - order (ForeignKey to Order),
 - product (ForeignKey to Product),
 - quantity (PositiveIntegerField)

API Endpoints:

- **Customers:**
 - List all customers: **GET /api/customers/**
 - Create a new customer: **POST /api/customers/**
 - Update exiting customer: **PUT /api/customers/<id>/**
- **Products:**
 - List all products: **GET /api/products/**
 - Create a new product: **POST /api/products/**

- **Orders:**
 - List all orders: **GET /api/orders/**
 - Create a new order with multiple product and corresponding quantity: **POST /api/orders/**
 - Edit existing order: **PUT /api/orders/<id>/**
 - List order based on the products: **GET /api/orders/?products=Book,Pen**
 - List order based on the customer: **GET /api/orders/?customer=Sam**

Validations (This must be in serializers)

- Customer's and product's name must be unique
- Weight in positive decimal and not more than 25kg
- Order cumulative weight must be under 150kg
- Order Date cannot be in past.

Example Payload of Order Creation:

```
{
  "customer": 2,
  "order_date": "31/12/2023",
  "address": "Test Address",
  "order_item": [
    {
      "product": 1,
      "quantity": 1,
    },
    {
      "product": 7,
      "quantity": 2,
    },
    {
      "product": 4,
      "quantity": 3,
    }
  ]
}
```