**Day-2**

**Objective**:

The purpose of this code is to define basic data structures for e-commerce system. These schemas model products, orders, and delivery zones

**Product:**

Details of products (like id, name, price, stock, category, etc.) can be stored and managed.

**Order:**

Customer orders and their details (such as customer information, order items, status, total amount) can be tracked.

**Delivery Zone:**

Delivery zones and their drives can be managed to ensure smooth delivery operations.

Key Points From the code (Schemas).

**Product Schema:**

**Id**: Unique Identifier for the product.

**Name**: Name of Product.

**Price**: Price of Product.

**Stock**: Number of items available in stock.

**Category**: Product Category (can ‘Women’, ‘Men’, or ‘Kids’).

**Size**: (optional) Available Size (‘S’, ‘M’, ‘L’, ‘XL’).

**Color**: (optional)Available color.

**Description**: (optional) Product Description.

**ImageURL**: (optional) URL for the product image.

**Product API Endpoints**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Endpoint** | **Description** | **Status Codes** |
| GET | /api/products | List all products | 200 (OK), 500 (Internal Error) |
| GET | /api/products/{id} | Get product by ID | 200 (OK), 404 (Not Found) |
| POST | /api/products | Create a new | 201 [Created],400 (Bad request |
| PUT | /api/products/{id} | Update product by ID | 200 (OK), 400 (Bad Request) 404 Not Found |
| DELETE | /api/products/{id} | Delete product by ID | 200 (OK), 404 Not Found |

**Order Schema:**

CustomerInfo: Includes Customer detail like name, email, phone, (optional), and address.

OrderItem: Represents a Single Product in the order, with fields like productID, quantity, and price.

**Order:**

orderId: Unique order identifier.

customerInfo: Customer’s details.

orderDetail: List of items orderd.

Status: Order Status (e.g, ‘Pending’, ‘Processing’, ‘Shipped’, ‘Delivered’,’Cancelled’.)

orderDate: Date when the order was placed.

Totalamount: Total amount of the order.

**Orders API Endpoints**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Endpoint** | **Description** | **Status Codes** |
| GET | /api/orders | List all orders | 200 (OK), 500 (Internal Error) |
| GET | /api/ orders /{id} | Get orders by ID | 200 (OK), 404 (Not Found) |
| POST | /api/ orders | Create a new orders | 201 [Created],400 (Bad Request) |
| PUT | /api/ orders /{id} | Update orders by ID | 200 (OK), 400 (Bad Request) 404 Not Found |
| DELETE | /api/ orders /{id} | Delete orders by ID | 200 (OK), 404 Not Found |

**Delivery Zone Schema:**

CoverageArea: Specifies a city and list of postal codes covered by a delivery zone.

AssignedDriver: Information about a delivery driver, including driverid, name, and contact

**DeliveryZone:**

**zoneName**: Name of the delivery zone.

coverageArea: Area Covered by the delivery zone.

assignedDrivers: List of drivers assigned to this zone.

**Delivery API Endpoints**

|  |  |  |  |
| --- | --- | --- | --- |
| **Method** | **Endpoint** | **Description** | **Status Codes** |
| GET | /api/deliveryZones | List all deliveryZones | 200 (OK), 500 (Internal Error) |
| GET | /api/ deliveryZones /{id} | Get deliveryZones by ID | 200 (OK), 404 (Not Found) |
| POST | /api/ deliveryZones | Create a new deliveryZones | 201 [Created],400 (Bad Request) |
| PUT | /api/ deliveryZones /{id} | Update deliveryZones by ID | 200 (OK), 400 (Bad Request) 404 Not Found |
| DELETE | /api/ deliveryZones /{id} | Delete deliveryZones by ID | 200 (OK), 404 Not Found |