/.
Slash

Backend Intern Task

Task: Implement an Order Management System

Objective:

Design and implement an Order Management System (OMS) for our e-commerce mobile app.

Requirements:

- **Environment Setup**:
  - Ensure you have a working environment with the necessary tools and technologies.
  - **Backend Framework**: NestJS
  - **ORM**: Prisma
  - **Database**: PostgreSQL

- **Database Schema**:
  - Create the following entities:
    - **Users**: `userId, name, email, password, address,` etc
    - **Products**: `productId, name, description, price, stock,` etc
    - **Orders**: `orderId, orderDate, status, etc`
    - **Cart:** `cartId, etc`

- **API Endpoints**:
  - **Add to Cart:**
    - **Endpoint**: `POST /api/cart/add`
    - **Functionality**: Adds a product to the user's cart or updates the quantity if the product is already in the cart.
  - **View Cart**:
    - **Endpoint**: `GET /api/cart/:userId`
    - **Functionality**: Retrieves the user's cart.
  - **Update Cart**:
    - **Endpoint**: `PUT /api/cart/update`

- ■ **Functionality**: Updates the quantity of a product in the cart.
    - ○ **Remove From Cart**:
        - ■ **Endpoint**: DELETE /api/cart/remove
        - ■ **Functionality**: Removes a product from the cart.
    - ○ **Create Order**:
        - ■ **Endpoint**: POST /api/orders
        - ■ **Functionality**: Creates a new order for the specified user with the products in their cart.
    - ○ **Get Order by ID**:
        - ■ **Endpoint**: GET /api/orders/:orderId
        - ■ **Functionality**: Retrieves the order details by order ID.
    - ○ **Update Order Status**:
        - ■ **Endpoint**: PUT /api/orders/:orderId/status
        - ■ **Functionality**: Updates the status of an order.

- **Documentation**:
    - ○ Document your code and API endpoints using Swagger or a similar tool.
    - ○ Provide a README file with instructions on how to set up, run, and test your code.

Submission:

- Create a public repository on github
- Implement the above task in your branch.
- Provide a README file with instructions on how to set up, run, and test your code.
- Submit your work through this form: https://forms.gle/Qa3zBFKwqmHRLuGc8

Evaluation Criteria:

- **Code Quality**: Clean, maintainable, and well-documented code.
- **Functionality**: Correct implementation of the Order Management System according to the requirements.
- **Documentation**: Clear and concise documentation.

Bonus Points:

- **Video Explanation:**

    Interns who record a video explaining their decisions and the implementation process will receive bonus points. The video should cover the following:

- Overview of the implemented features.
- Explanation of the technology stack and design decisions.
- Demonstration of how to set up, run, and test the application.
- **Features:**

  -Implement a feature for order history retrieval:

  - **Endpoint**: `GET /api/users/:userId/orders`

  -Implement a feature for applying discounts and coupons to orders:

  - **Endpoint**: `POST /api/orders/apply-coupon`

Good luck, and we look forward to your implementation!