📘 API Documentation

General Information

Version: v0

Base URL: http://localhost:8080

Format: Follows OpenAPI 3.0 Specification

📂 Category Controller

🔹 GET `/categories/{id}`

Description: Get a category by ID.

Parameters:

- `id` (path, integer, required): The ID of the category.

Responses:

- `200 OK`: Returns a `CategoryDTO` object.

🔹 PUT `/categories/{id}`

Description: Update an existing category by ID.

Parameters:

- `id` (path, integer, required): The ID of the category.

Request Body:

- `CategoryDTO`

Responses:

- `200 OK`: Updated `CategoryDTO` object.

🔹 DELETE `/categories/{id}`

Description: Delete a category by ID.

Parameters:

- `id` (path, integer, required): The ID of the category.

Responses:

- `200 OK`

🔹 GET `/categories`

Description: Retrieve all categories.

Responses:

- `200 OK`: List of `CategoryDTO` objects.

🔹 POST `/categories`

Description: Create a new category.

Request Body:

- `CategoryDTO`

Responses:

- `200 OK`: Created `CategoryDTO` object.

📦 Product Controller

🔹 POST `/products/createProduct`

Description: Create a new product.

Request Body:

- `CreateProductDTO`

Responses:

- `200 OK`: Created `ProductDTO` object.

🔹 PATCH `/products/updateProduct/{id}`

Description: Update an existing product by ID.

Parameters:

- `id` (path, integer, required): The ID of the product.

Request Body:

- Partial properties of `ProductDTO`

Responses:

- `200 OK`: Updated `ProductDTO` object.

🔹 GET `/products/product/{id}`

Description: Get a product by ID.

Parameters:

- `id` (path, integer, required): The ID of the product.

Responses:

- `200 OK`: Returns `ProductDTO` object.

🔹 GET `/products/product/all`

Description: Retrieve all products.

Responses:

- `200 OK`: List of `ProductDTO` objects.

🛒 Order Controller

🔹 POST `/orders/place`

Description: Place an order.

Responses:

- `200 OK`: Returns `OrderDTO` object.

🔹 POST `/orders/confirmPayment/{paymentOrderId}`

Description: Confirm order payment.

Parameters:

- `paymentOrderId` (path, string, required): Payment order ID.

Responses:

- `200 OK`: Updated `OrderDTO` object.

🔹 POST `/orders/cancel/{orderId}`

Description: Cancel an order.

Parameters:

- `orderId` (path, integer, required): Order ID.

Responses:

- `200 OK`: Cancelled `OrderDTO` object.

🔹 GET `/orders`

Description: Get all orders.

Parameters:

- `status` (query, string, optional): Filter by order status.

Responses:

- `200 OK`: List of `OrderDTO` objects.

🔹 GET `/orders/{orderId}`

Description: Get order by ID.

Parameters:

- `orderId` (path, integer, required): Order ID.

Responses:

- `200 OK`: Returns `OrderDTO` object.

🧺 Cart Controller

🔹 GET `/carts/cart`

Description: Retrieve the current cart.

Responses:

- `200 OK`: `CartResponseDTO` object.

🔹 POST `/carts/cart`

Description: Add items to the cart.

Request Body:

- `CartItemDTO`

Responses:

- `200 OK`: Updated `CartResponseDTO` object.

🔹 DELETE `/carts/cart`

Description: Clear the cart.

Responses:

- `200 OK`

🔹 PATCH `/carts/cart`

Description: Update cart item quantity.

Parameters:

- `cartItemId` (query, integer, required): Cart item ID.
- `quantity` (query, integer, required): New quantity.

Responses:

- `200 OK`: Updated `CartResponseDTO` object.

🔹 PATCH `/carts/cart/removeItem/{cartItemId}`

Description: Remove an item from the cart.

Parameters:

- `cartItemId` (path, integer, required): Cart item ID.

Responses:

- `200 OK`: Updated `CartResponseDTO` object.

📘 Schemas

🗒️ CategoryDTO

Field	Type	Description
id	integer	Unique category ID
name	string	Name of the category

🗒️ CreateProductDTO

Field	Type	Description
title	string	Product title

description	string	Detailed product description
price	double	Price of the product
stockQuantity	integer	Available stock quantity
rating	double	Product rating (e.g., 4.5)
category	string	Name of associated category

🗒️ ProductDTO

Field	Type	Description
id	integer	Unique product ID
title	string	Product title
description	string	Product description
price	double	Price of the product
stockQuantity	integer	Quantity in stock
rating	double	Product rating
category	string	Category of the product

🗒️ OrderDTO

Field	Type	Description
orderId	integer	Unique order ID
userId	integer	ID of the user who placed order
status	string	Current status of the order
orderDate	date-time	Date and time of order placement
paymentMethod	string	Method used for payment
paymentStatus	string	Status of the payment
totalAmount	double	Total cost of the order
items	array	List of ordered items (OrderItemsDTO)

🗒️ CartItemDTO

Field	Type	Description
productId	integer	ID of the product
quantity	integer	Quantity to be added

🗒️ CartResponseDTO

Field	Type	Description
id	integer	Cart ID
userId	integer	ID of the user
items	array of CartItemResponseDTO	List of cart items
totalPrice	double	Total price of items in the cart

✅ Conclusion

This document outlines all REST API endpoints, request and response structures, and schema definitions for the `v0` version of your service. You can use this to generate Swagger/OpenAPI UI or integrate with frontend/backend teams seamlessly.