Category API:

This part of the system handles categories, and each category can have multiple products. The API allows creating categories, updating categories, retrieving categories (both single and all), and deleting categories.

1. Create Category with Multiple Products (POST /api/categories):

You can create a new category and assign multiple products to it in a single request. The request body contains the category's name and the list of products.

Request:

```
POST http://localhost:8080/api/categories
Content-Type: application/json
```

Body:

```
{
  "categoryName": "Electronics",
  "products": [
    {
      "id": 1,
      "name": "Smartphone",
      "qty": 2,
      "price": 20000
    },
    {
      "id": 2,
      "name": "Laptop",
      "qty": 2,
      "price": 60000
    },
    {
      "id": 3,
      "name": "Tablets",
      "qty": 2,
      "price": 60000
    }
  ]
}
```

- This request will create a category "Electronics" with products like "Smartphone", "Laptop", and "Tablets".

Request Body for Books Category:

```
{
  "categoryName": "Books",
  "products": [
    {
      "id": 4,
      "name": "Java Programming Language",
      "qty": 2,
      "price": 2000
    },
    {
      "id": 5,
      "name": "Python Programming Language",
      "qty": 2,
      "price": 60000
    },
    {
      "id": 6,
      "name": "DSA",
      "qty": 2,
      "price": 60000
    }
  ]
}
```

- This request will create a category "Books" with products like "Java Programming Language", "Python Programming Language", and "DSA".

---

2. Get All Categories (GET /api/categories?page=0&size=10):

To retrieve all categories in a paginated way, you can use this endpoint.

Request:

```
GET http://localhost:8080/api/categories?page=0&size=10
```

- `page`: The page number you want to fetch (starting from 0).
- `size`: The number of categories per page (10 in this case).

---

3. Get Category by ID (GET /api/categories/{id}):

This endpoint allows you to retrieve a specific category by its ID.

Request:

```
GET http://localhost:8080/api/categories/1
```

- This will return the details of the category with ID `1`.

---

4. Update Category by ID (PUT /api/categories/{id}):

You can update a category's details (e.g., the category name or associated products) using this endpoint.

Request:

```
PUT http://localhost:8080/api/categories/1
Content-Type: application/json
```

Body:

```
{
  "categoryName": "Updated Electronics",
  "products": [
    {
      "id": 1,
      "name": "Smartphone",
      "qty": 3,
      "price": 22000
    },
    {
      "id": 2,
      "name": "Laptop",
      "qty": 3,
      "price": 65000
    }
  ]
}
```

- This will update the category "Electronics" (ID: 1) and modify the product details.

---

5. Delete Category by ID (DELETE /api/categories/{id}):

To delete a category, use this endpoint with the category's ID.

Request:

```
DELETE http://localhost:8080/api/categories/1
```

- This will delete the category with ID `1` and its associated products.

Product API:

The product management API allows creating, updating, retrieving, and deleting individual products.

---

1. Create Product (POST /api/products):

You can add a new product using this endpoint by providing product details such as `id`, `name`, `quantity`, and `price`.

Request:

```
POST http://localhost:8080/api/products
Content-Type: application/json
```

Body:

```
{
  "id": 3,
  "name": "Camera",
  "qty": 2,
  "price": 20000
}
```

- This will create a product named "Camera" with 2 units, priced at 20,000.

---

2. Get All Products (GET /api/products?page=0&size=10):

You can retrieve all products in a paginated format using this endpoint.

Request:

```
GET http://localhost:8080/api/products?page=0&size=10
```

- Similar to categories, this allows you to fetch products with pagination.

---

3. Get Product by ID (GET /api/products/{id}):

This endpoint allows you to retrieve product details by its ID.

Request:

```
GET http://localhost:8080/api/products/2
```

- This will return the product with ID `2` (e.g., "Laptop").

---

4. Update Product by ID (PUT /api/products/{id}):

You can update a specific product using this endpoint.

Request:

```
PUT http://localhost:8080/api/products/2
Content-Type: application/json
```

Body:

```
{
  "id": 2,
  "name": "Updated Laptop",
  "qty": 3,
  "price": 70000
}
```

- This will update the product with ID `2` (Laptop) to have 3 units and a new price of 70,000.

---

5. Delete Product by ID (DELETE /api/products/{id}):

You can delete a product using its ID.

Request:

```
DELETE http://localhost:8080/api/products/2
```

- This will delete the product with ID `2`.

Explanation:

- **Category API**: Allows CRUD operations for categories and their associated products.
- **Product API**: Handles CRUD operations for individual products.
- **Pagination**: Both Category and Product APIs support pagination (`page`, `size`) to retrieve a subset of records efficiently.
- **POST**: Used to create new categories and products.
- **PUT**: Used to update existing categories and products.
- **GET**: Used to fetch categories/products either individually or in bulk.
- **DELETE**: Used to remove categories or products.