

Node.js CRUD Application with MongoDB

Node Task - 01

Task Overview

This task is a simple CRUD (Create, Read, Update, Delete) application built using Node.js, Express.js, and MongoDB. This task aimed to demonstrate backend development skills, including setting up an Express server, connecting to a MongoDB database, defining a data schema, and handling API requests.

Technologies Used

- **Node.js** – JavaScript runtime environment
- **Express.js** – Web framework for Node.js
- **MongoDB** – NoSQL database
- **Mongoose** – ODM (Object Data Modeling) library for MongoDB
- **Postman** – API testing tool
- **Git & GitHub** – Version control and code hosting

Product Schema


A product is represented with the following schema:

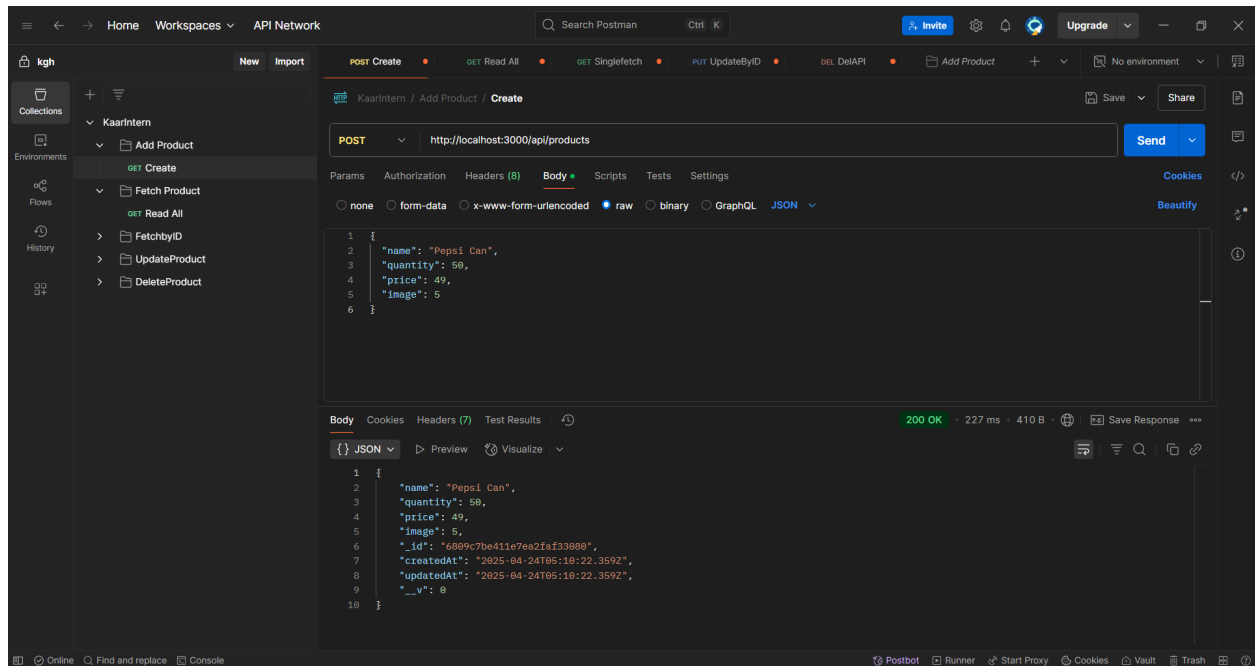
```
{
  name: String,           // Required
  quantity: Number,       // Required, default: 0
  price: Number,          // Required, default: 0
  image: Number           // Optional, default: 0
}
```

API Endpoints

GET /api/products


Retrieve all products.

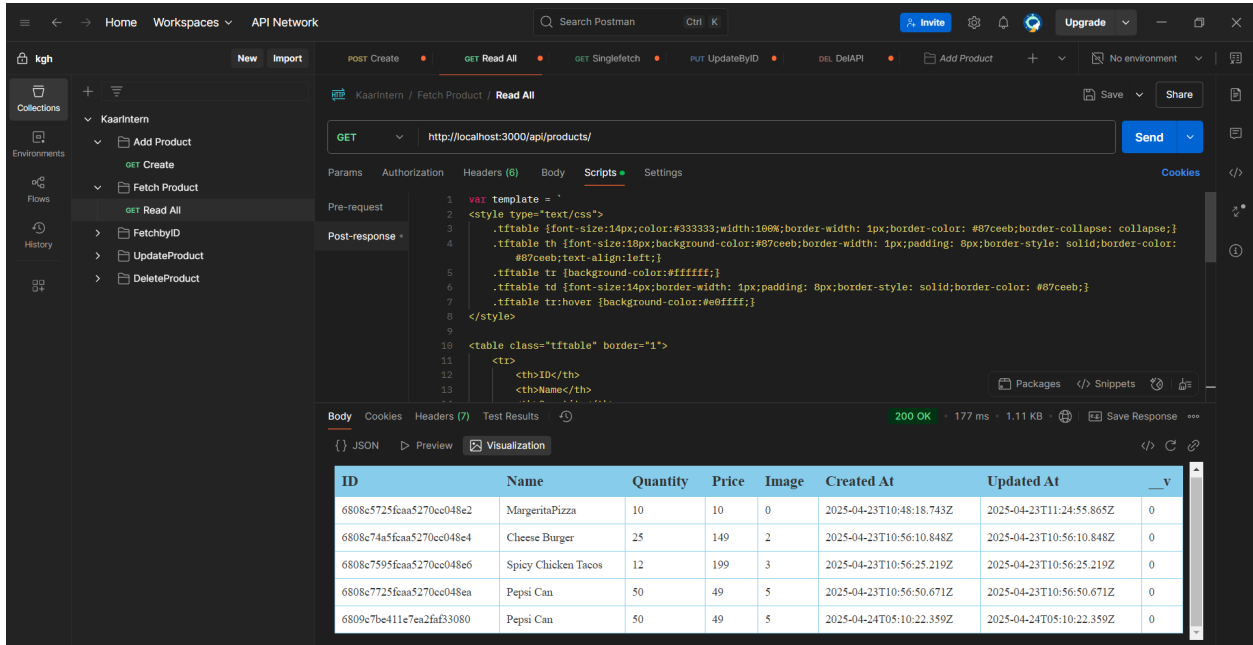
 Screenshot:



GET /api/products/:id

Retrieve a product by ID.

 Screenshot:




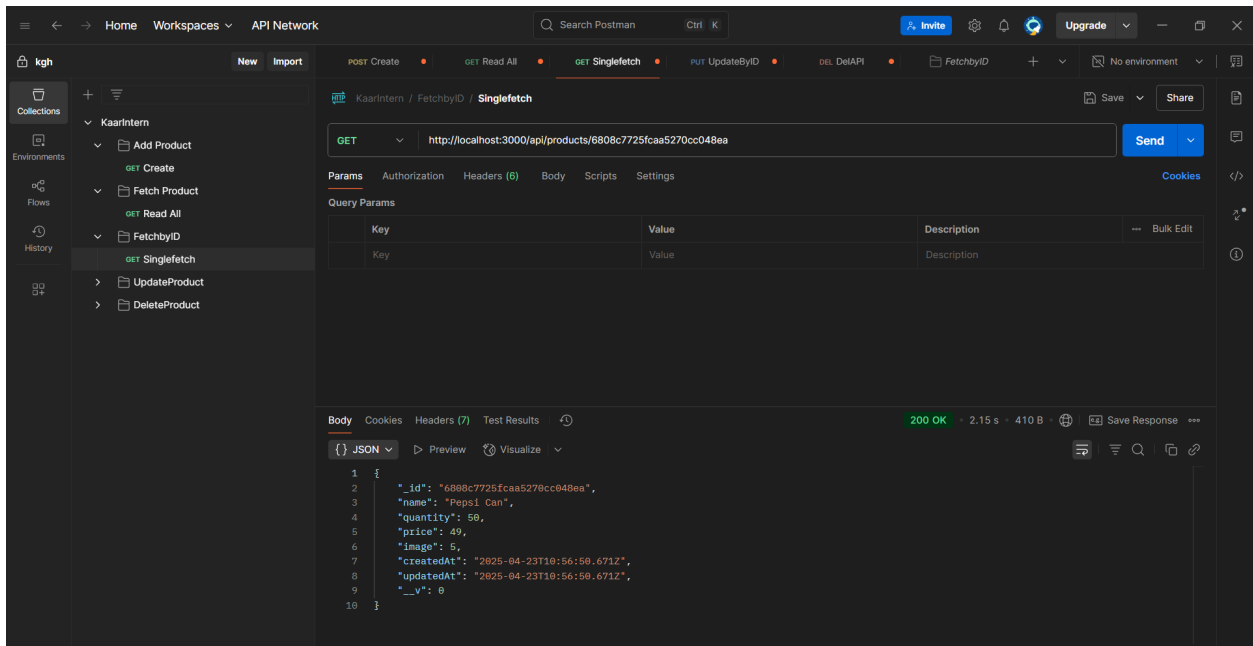
The screenshot shows a Postman interface with a GET request to `http://localhost:3000/api/products/`. The response is a 200 OK status with a response time of 177 ms and a body size of 1.11 KB. The response body is a JSON array of product objects, which is visualized as a table.

ID	Name	Quantity	Price	Image	Created At	Updated At	_v
6808c5725fcaa5270cc048e2	MargheritaPizza	10	10	0	2025-04-23T10:48:18.743Z	2025-04-23T11:24:55.865Z	0
6808c74a5fcaa5270cc048e4	Cheese Burger	25	149	2	2025-04-23T10:56:10.848Z	2025-04-23T10:56:10.848Z	0
6808c7595fcaa5270cc048e6	Spicy Chicken Tacos	12	199	3	2025-04-23T10:56:25.219Z	2025-04-23T10:56:25.219Z	0
6808c7725fcaa5270cc048ea	Pepsi Can	50	49	5	2025-04-23T10:56:50.671Z	2025-04-23T10:56:50.671Z	0
6809c7be411e7ea2faf33080	Pepsi Can	50	49	5	2025-04-24T05:10:22.359Z	2025-04-24T05:10:22.359Z	0

POST /api/products


Create a new product.

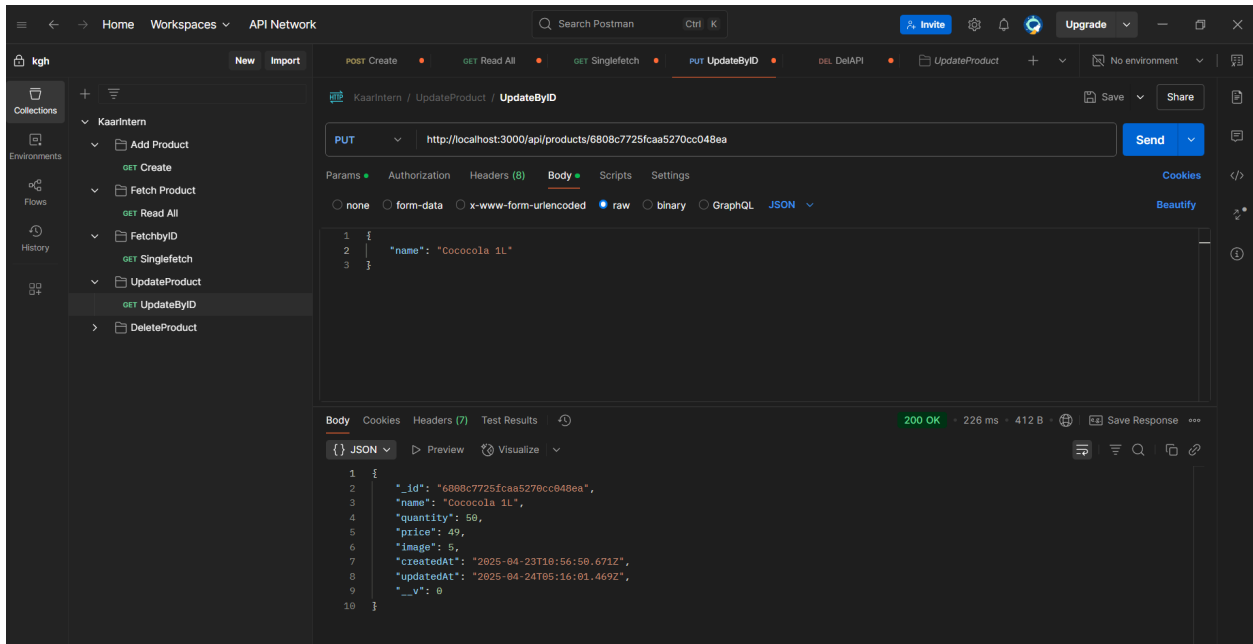
 Screenshot:



PUT /api/products/:id


Update a product by ID.

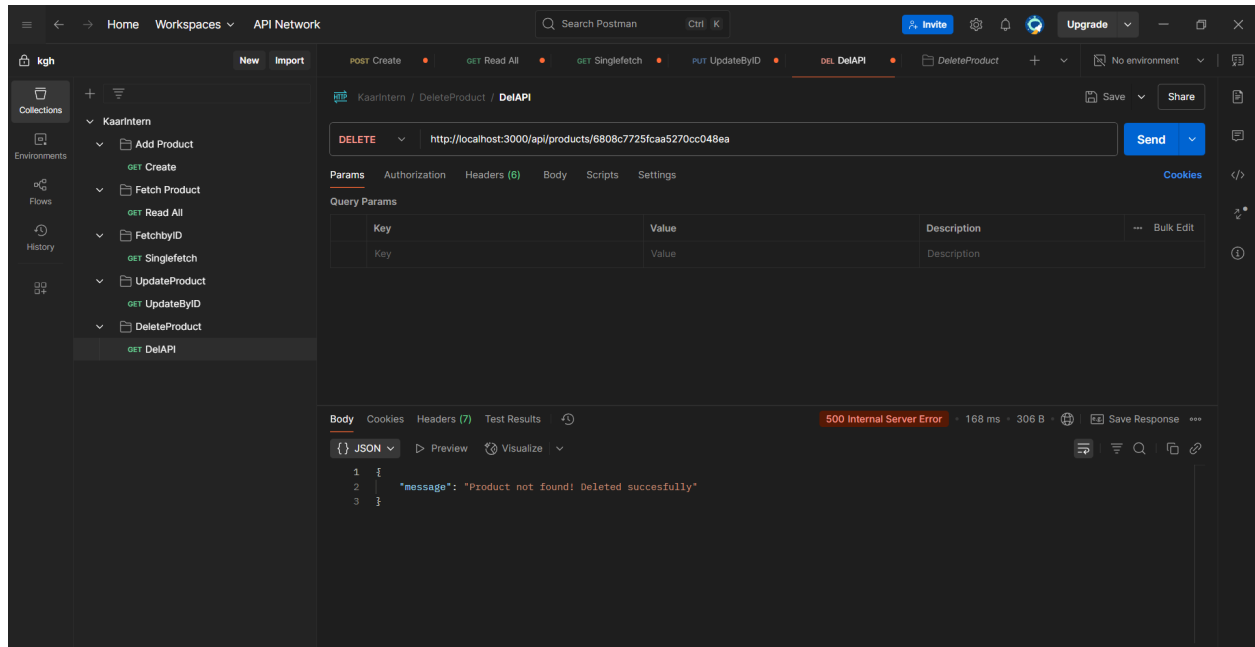
 Screenshot:



DELETE /api/products/:id

Delete a product by ID.

 **Screenshot:**



Version Control

- Git was initialized.
- Commits were made at major development milestones.
- Project pushed to GitHub for collaboration and review.

 **GitHub Repository:**

https://github.com/SuryaCreatX/Kaar_NodeLearn.git