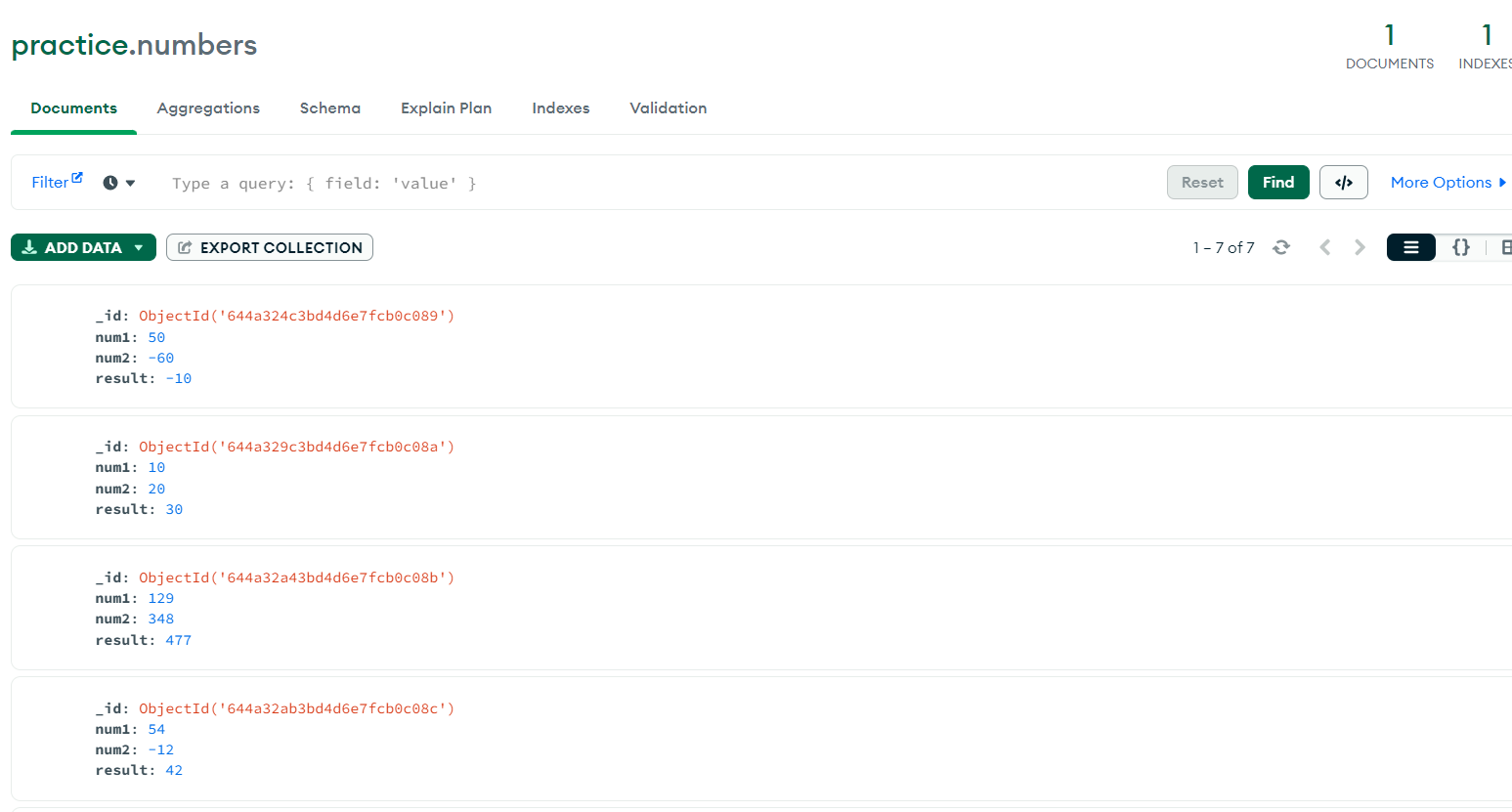
**Documentation of API**

**Mongodb:**



In mongodb you will get that data stored in the database named as mydata as shown in above snap.

**Introduction**

This code is a simple example of a Flask API that performs CRUD (Create, Read, Update, and Delete) operations on a MongoDB database using the PyMongo library.

**Prerequisites**

Before running this code, you need to make sure you have Python, Flask, PyMongo, and MongoDB installed on your machine.

**Installation**

You can install Flask, PyMongo, and other required libraries by running the following command in your terminal:

pip install Flask pymongo

pip install Flask

**Usage**

To run this code, navigate to the directory where the file is located in your terminal, and run the following command:

python filename.py

Once the server is running, you can access the API using a web browser or a tool like Postman.

**Endpoints**

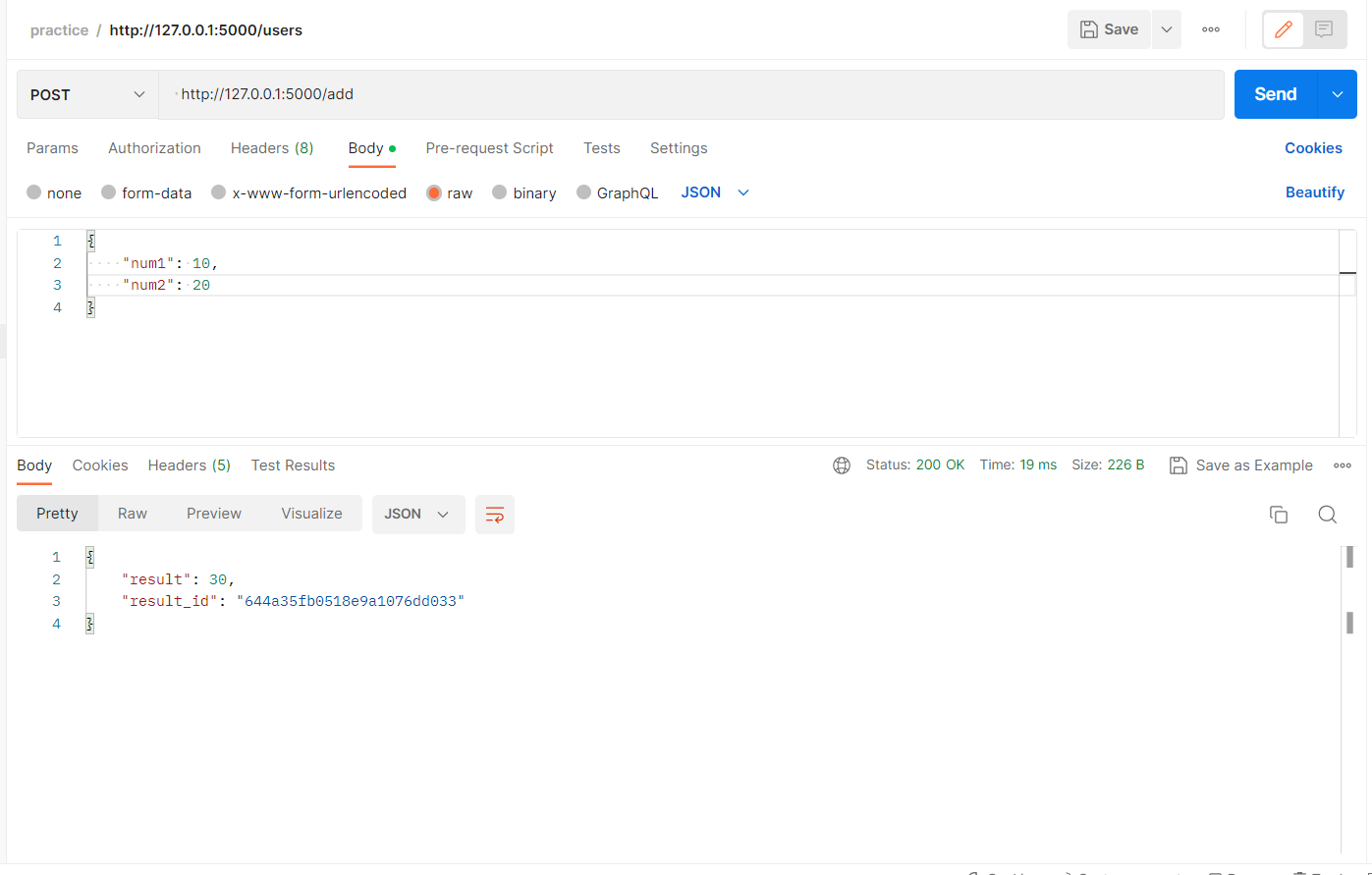
**GET /**

This endpoint returns a simple welcome message.

**POST /add**

<http://127.0.0.1:5000/add>

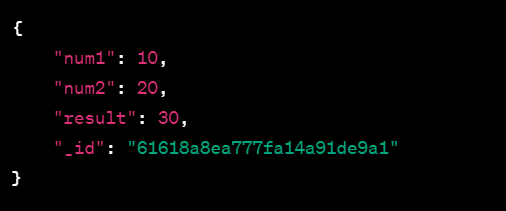
This endpoint accepts a JSON payload with num1 and num2 values, adds them together, and inserts the result into the MongoDB database. It then returns a JSON response with the calculated result.



**GET /get/<id>**

[**http://localhost:5000/get/61618a8ea777fa14a91de9a1**](http://localhost:5000/get/61618a8ea777fa14a91de9a1)

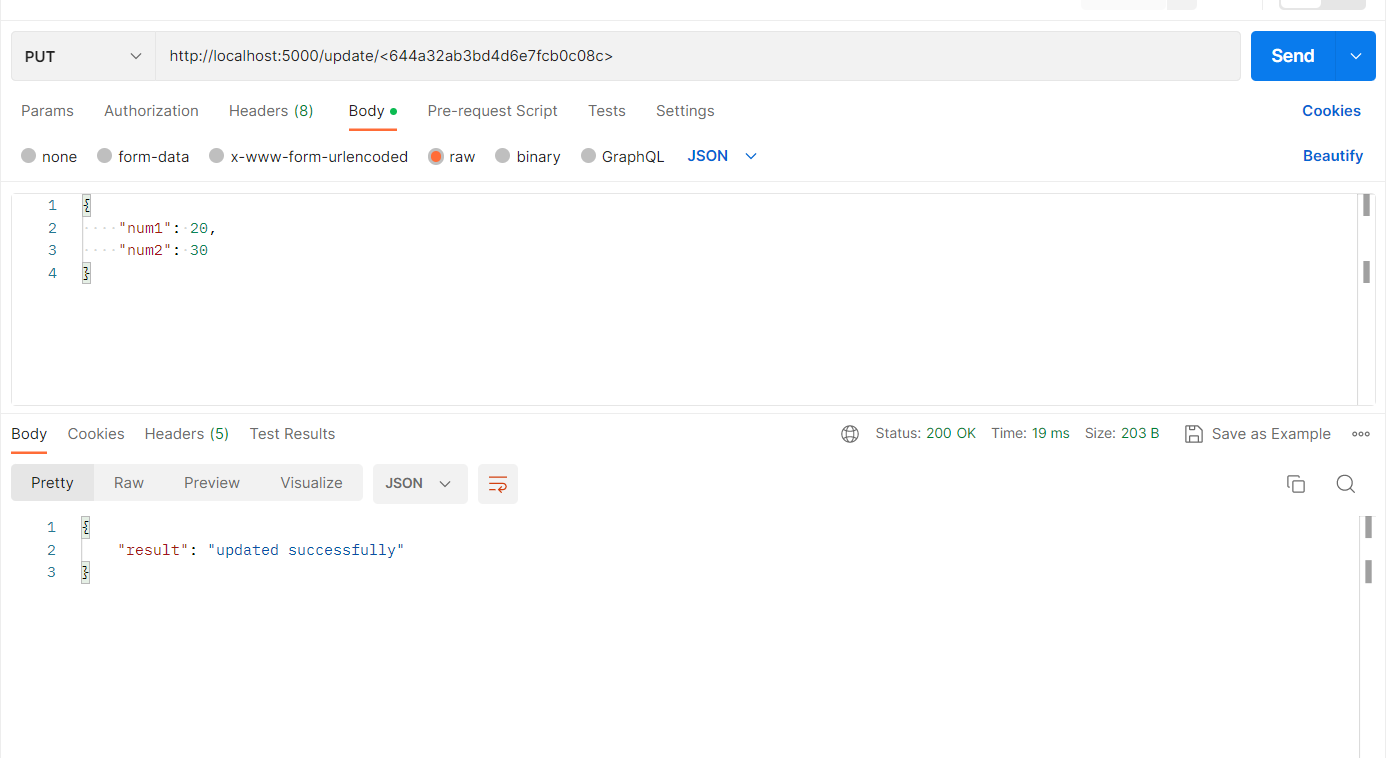
This endpoint accepts an ID parameter and retrieves the document with that ID from the MongoDB database. It then returns a JSON response with the document's num1, num2, and result values.



**PUT /update/<id>**

[http://localhost:5000/update/<644a3542a0681100e22ed9ee>](http://localhost:5000/update/%3c644a3542a0681100e22ed9ee%3e)

This endpoint accepts an ID parameter and a JSON payload with new num1 and num2 values. It updates the corresponding document in the MongoDB database with the new values and recalculates the result. It then returns a JSON response with the updated result.



**DELETE /delete/<id>**

**Endpoints:-** [http://localhost:5000/delete/<644a329c3bd4d6e7fcb0c08a>](http://localhost:5000/delete/%3c644a329c3bd4d6e7fcb0c08a%3e)

This endpoint accepts an ID parameter and deletes the corresponding document from the MongoDB database. It then returns a JSON response confirming the deletion.

