Using Postman to Test CRUD Operations for Flask + MongoDB

This guide shows how to test Create, Read, Update, and Delete operations using Postman with your Flask app and MongoDB database.

Prerequisites

1. Start your Flask app by running 'python app.py'. Make sure it is running at http://127.0.0.1:5000/

2. Download and install Postman from https://www.postman.com/downloads/

API Endpoint Overview

Here are the endpoints you'll test:

- POST /add: Add a new animal to the database
- GET /animals: Get all animal records
- GET /animals/<animal_id>: Get a specific animal
- PUT /animals/<animal_id>: Update an existing animal
- DELETE /animals/<animal_id>: Delete an animal

CRUD Operations

1. Create a New Animal

Method: POST

URL: http://127.0.0.1:5000/add

Headers: Content-Type: application/json

Body (raw JSON): Click "Body" → "raw" → choose JSON from the dropdown and add the following:

```
{
  "animal_id": "A123456",
  "name": "Baxter",
  "animal_type": "Dog",
  "breed": "Labrador Retriever",
  "color": "Black",
  "age_upon_outcome_in_weeks": 52,
  "outcome_type": "Adoption"
}
```

Click Send. You should get a success message.

2. Read All Animals

Method: GET

URL: http://127.0.0.1:5000/animals

Click Send. You should get a list of all animals.

3. Read One Animal

Method: GET

URL: http://127.0.0.1:5000/animals/A123456

Click Send. If found, the animal's data is returned. If not, an error message appears.

4. Update an Animal

Method: PUT

URL: http://127.0.0.1:5000/animals/A123456

Headers: Content-Type: application/json

Body (raw JSON):

```
{
  "color": "Black and White",
  "outcome_type": "Transfer"
}
```

Click Send. A confirmation message is returned.

5. Delete an Animal

Method: DELETE

URL: http://127.0.0.1:5000/animals/A123456

Click Send. A success or error message will appear.

Summary

Postman allows you to test your backend endpoints quickly and efficiently without a frontend interface. It helps validate logic, structure, and database interaction.