## About the Project/Project Title

This project reads, creates, updates, and deletes different animals in an animal database that uses MongoDB. The purpose of this project is to be able to access and work with the database using Python. The user should be able to add and search for different animals in the database. The user should also be able to update information about an existing animal in the database and delete an animal from the database.

## Motivation

This project uses Dash, Python, and MongoDB to practice creating CRUD functions and working with a database.

## Getting Started

To get started, load the ProjectTwoDashboard.ipynb file, main.ipynb file, and CRUDMongo.py file.

## Installation

Jupyter Notebook – install from <https://jupyter.org/install>

MongoDB – install from https://www.mongodb.com/docs/manual/installation/

## Usage

### Code Example

Create:

new\_animal = {

“name” = “Bill”,

“age\_upon\_outcome” = “3 years”,

.

.

.

}

result = pets.create(new\_animal)

Read:

records = animal\_shelter.read({“breed” : “Husky”})

Update:

Update\_result = pets.collection.update\_many({“breed” : ”Siamese”}, {“breed” : “Shorthair”})

Delete:

Delete\_result = pets.collection.delete)many({“breed” : “Siamese”})

### Tests

To test if my code worked, I made an ipynb (Jupyter Notebook) file that I ran tests with. I created a new animal and then called the create method on it to check if it worked. I also queried the database for a specific breed using the read method. To test the update method, I queried for a breed of animal and then changed the breed in the database. To test the delete method, I queried the database for a specific breed and deleted it from the database. For both the update and delete, I also checked to get feedback on how many animals were either updated or deleted.

### Screenshots

MongoDB Import:

A screenshot of a computer

Description automatically generated

User Authentication:

A screenshot of a computer program

Description automatically generated

Create and Read:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Update and Delete:

A screen shot of a computer code

Description automatically generated

Dashboard – Unique Identifier and Image/Logo

A screenshot of a computer

Description automatically generated

Dashboard - Filter:

A screenshot of a computer

Description automatically generated

## Tools

MongoDB was chosen as the database for this project because it is flexible and has a native driver for the Python language. MongoDB works well with web applications as well, and Python is good for JSON documents. These factors made it an excellent choice when building this project. Dash is used to build web applications, which was a goal of this project. Using Dash, I was able to display animals from the database and filter through it.

Dash:

<https://dash.plotly.com/installation>

MongoDB:

<https://www.mongodb.com/docs/manual/installation/>

Python:

<https://www.python.org/downloads/>

**Steps and Challenges**

After completing the main.ipynb and AnimalShelter.py files, I also had to complete the Dash file. To complete this, I had to give Dash access to AnimalShelter. I had to create an instance of AnimalShelter that could then be queried against to filter through. I also had to create a graph and map to display information from the database. A challenge I had was creating the buttons for filtering. To overcome this, I had to slowly and carefully look over my code and fix issues I found. I also had to do research using websites on Dash and Python to figure out where I was going wrong.

## Contact

Your name: Mao Christie