# README

## About the Project/Project Title

This project encompasses an AnimalShelter Python class and the associated Python test script to test the various methods in the class. The intent of the test script is to be used in Jupyter Notebooks. The AnimalShelter Python class is designed to be a tool for an international rescue-animal training company to manage all their animals for clients.

## Motivation

This project exists as part of the Computer Science curriculum from Southern New Hampshire University and is intended to teach the student how to administer MongoDB and to insert Python middleware to manipulate the database. The AnimalShelter class is a part of the functional middleware, and the test script is a means to test the middleware as it manipulates the database. This will allow Grazioso Salvare to successfully manage animals for their rescue-animal training company and the various clients that use this tool.

## Getting Started

To start, you will need access to SNHU’s Virtual Lab provided by apporto. The Virtual Lab will have an established environment to develop Python classes via Jupyter Notebook and Python 3.9. Critical components of this project are the Python interpreter which executes code from classes and test scripts, Jupyter Notebook which is a tool to create scripts and classes, and the MongoDB which serves as the repository for all the data used for the international rescue-animal training company.

## Installation

The Virtual Lab has installed these appropriate tools for you. The AnimalShelter class and test script may be uploaded to the ‘Documents’ folder. Once uploaded, the test script may be used by executing Jupyter Notebook and browsing to the test script and executing it.

## Usage

The AnimalShelter class establishes a MongoDB connection and has four methods, Create, Read, Update, and Delete*.* The purpose of these methods is to teach to the student the principles of CRUD.

### Code Example

### The AnimalShelter class establishes a connection to the existing MongoDB:

### def \_\_init\_\_(self):

### This class also writes and reads from the existing MongoDB after the connection is established:

### def create(self, data):

This method creates a new record from the data provided.

**def read(self, output=None):**

This method reads a record and prints a list from the data provided.

**def update(self, find=None, update=None):**

This method finds and updates a record from the data provided.

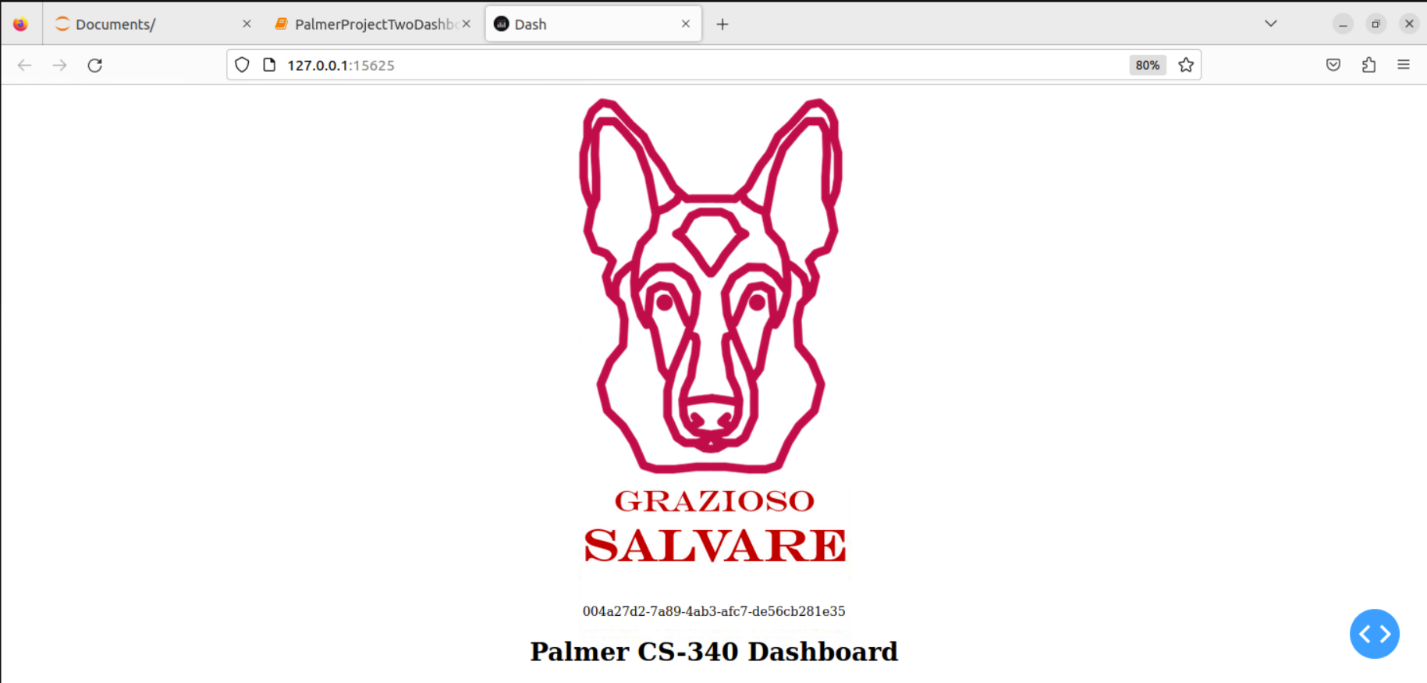
**def delete(self, output=None):**

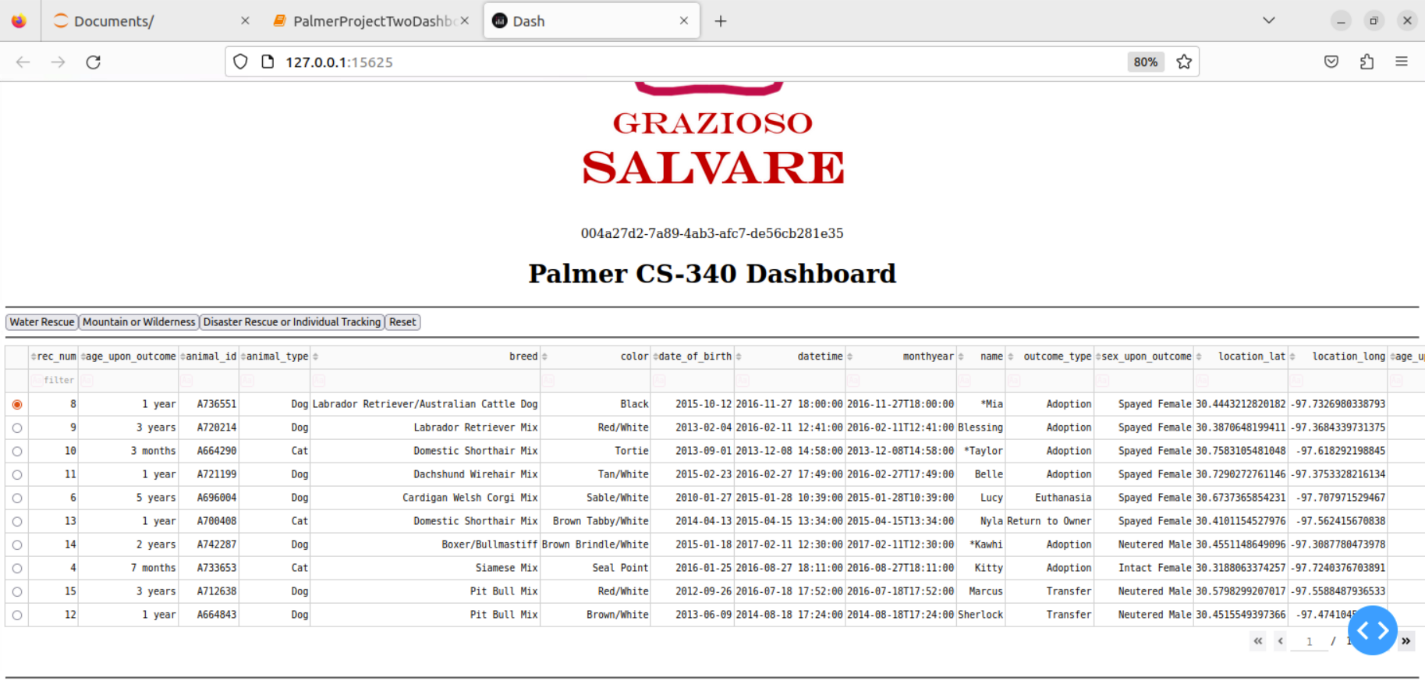
This method deletes a record from the data provided.

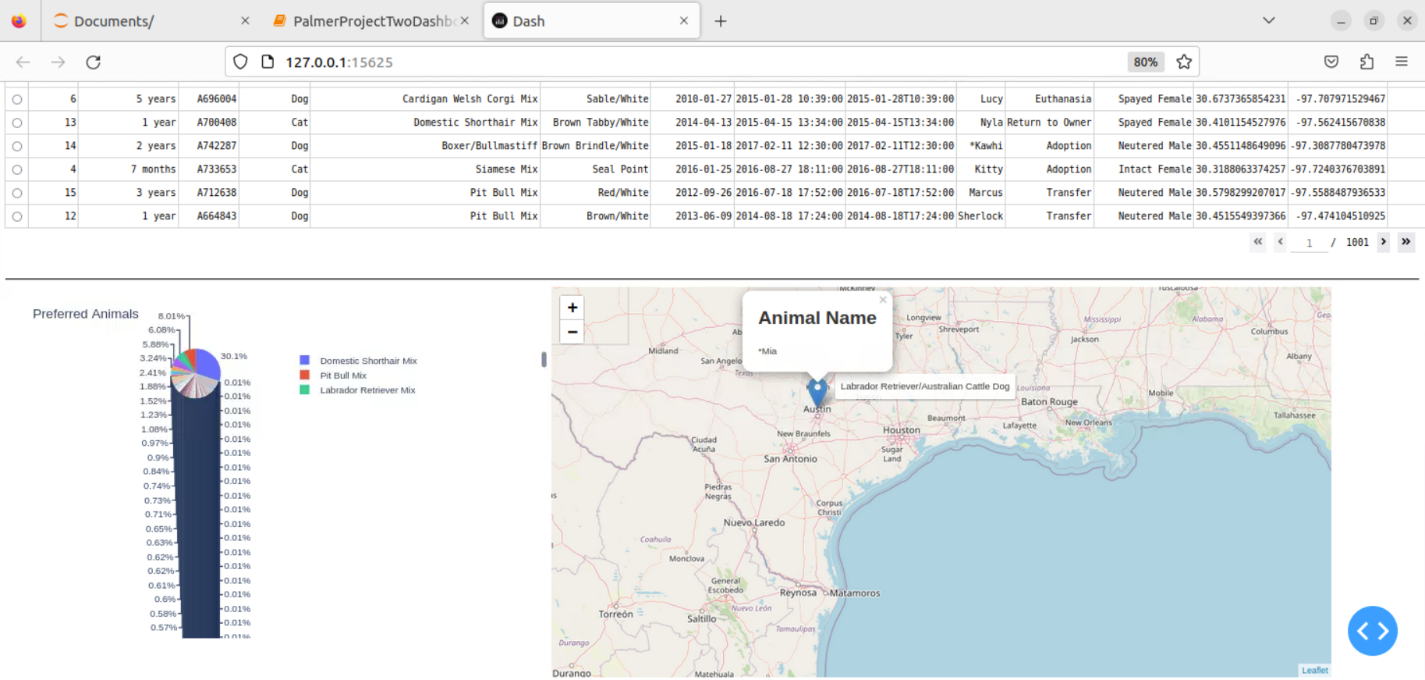
### Tests

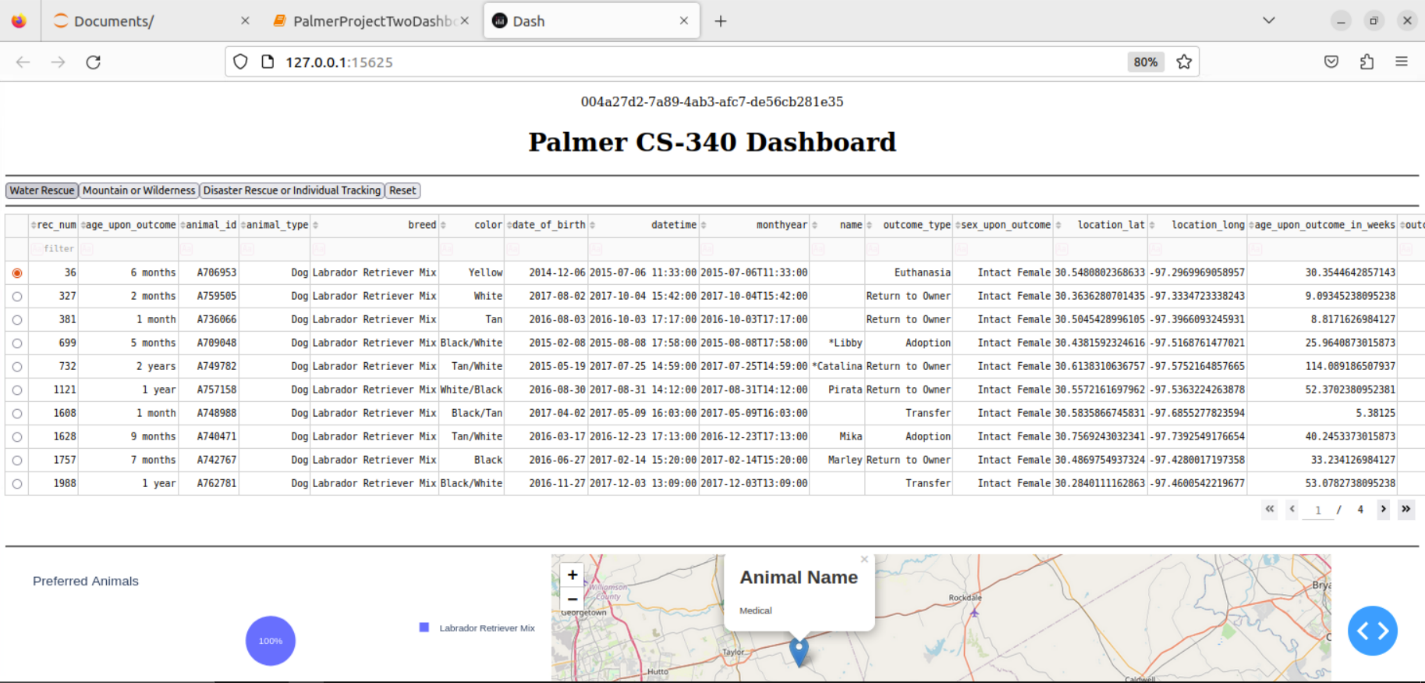
The associated test script creates an instance of AnimalShelter and employs the create, read, update and delete methods.

### Screenshots



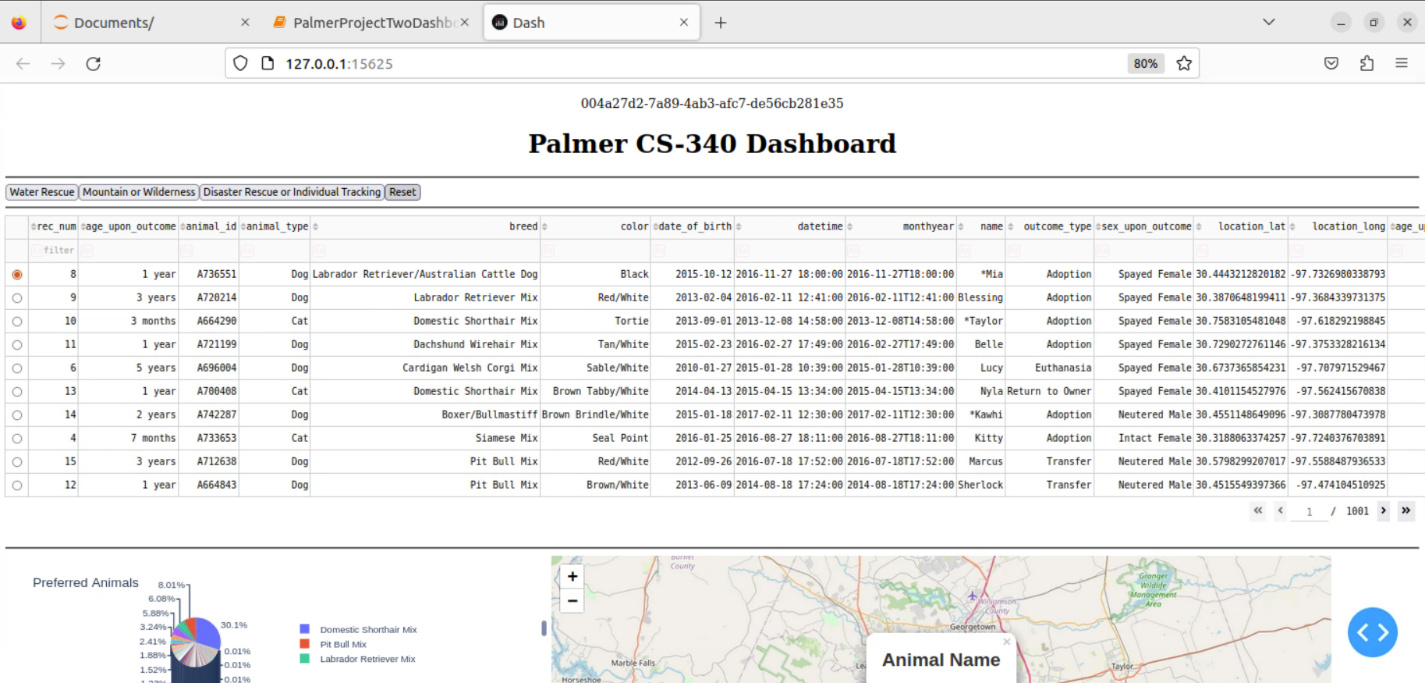




**

## 

## 



## Roadmap/Features (Optional)

*Provide an open issues list of proposed features (and known issues). If you have ideas for releases in the future, it is a good idea to list them in the README. What makes your project stand out?  
  
Note: This section is optional for the purposes of this assignment. If you choose not to fill out this section, remove it from your final README file.*

## Challenges

Most of the challenges encountered were a combination of developer errors such as indentation errors, comment errors, and improper method usage in the test script. Additional time was necessary to read about Python to successfully build scripts that tested the class properly.

## Contact

Your name: Alan Palmer