# CS 340 README

## About the Animal Shelter Python Module

The Animal Shelter Python module wraps Mongo database operations for the animal shelter database.

## Motivation

The module abstracts the underlying MongoDB connection and library use away from the rest of the application and provides operations for a particular database. The current implementation utilizes the "PyMongo" driver for synchronous access. A new implementation based on "Motor" could enable asynchronous database calls without requiring a drastic change to the application.

## Getting Started

Ensure the "**animal\_shelter.py**" file is in the same folder as your project.

from animal\_shelter import AnimalShelter

The add the following to the top of your python script.

## Installation

Prerequisites

*MongoDB with authentication enabled*

*Facts needed, username, password, server, port*

*Database "AAC"*

*Collection "animals"*

*"animals" populated from "aac\_shelter\_outcomes.csv" located in /usr/local/datasets/*

Python Environment

PyMongo module

$ python -m pip install 'pymongo'

## Usage

**Initialization**

User and Password are required. Host will default to "localhost". Port will default to "27017".

### Code Example

shelter = AnimalShelter("myUser", "myPassword", "localhost", "27017")

**Create**

The create method will insert a JSON document into the shelter collection.

### Code Example

shelter.create({ "animal\_id" : "A99999", "animal\_type" : "Cat", "breed" : "British Shorthair", "color" : "Black", "name" : "Soot" })

### Returns

*Returns True on success, or an exception on failure.*

### Screenshot

Graphical user interface

Description automatically generated with medium confidence

**Read One**

The read method can be used to return one or more documents. The find criteria determines the result. Search on a unique key and 1 record will be returned. A cursor will be returned in either case, but the iteration will be for 1 document.

### Code Example

shelter.read({ "animal\_id" : "A99999" })

### Returns

*Returns Cursor on success, otherwise an exception is thrown*

### Screenshot

Text

Description automatically generated

**Read Many**

The read method can also return multiple documents. The find criteria on a non-unique field will return multiple documents.

### Code Example

shelter.read({ "animal\_type" : "Cat" })

### Returns

*Returns Cursor on success, or an exception on failure.*

### Screenshot*Text, letter Description automatically generated*

**Update**

The update method will update or add fields. It is not a full document replacement. If the find criteria matches more than one document all documents matched will be changed.

### Code Example

shelter.update({ "animal\_id" : "A99999" }, { 'name': 'Smelly Cat' })

### Returns

*Returns PyMongo UpdateResult on success, or an exception on failure.*

### Screenshot

Text

Description automatically generated

**Delete**

The delete method will remove 1 or more documents from the collection.

### Code Example

shelter.delete({ "animal\_id" : "A99999" })

### Returns

*Returns PyMongo DeleteResult on success, or an exception on failure.*

**Screenshot**

## Graphical user interface, text Description automatically generated

**Module in Action**

**When the application runs, the default is to list all animals from the shelter database.**

**Text

Description automatically generated with low confidence**

**When a row is selected using the radial buttons on the left in the data table the map displays a marker for the shelter location of the animal. When the marker is selected it will show the animal's name (if available) and breed.**

**The pie chart represents the percentages each breed makes from the data table. Breed counts below a certain threshold are grouped under "Other breeds."**

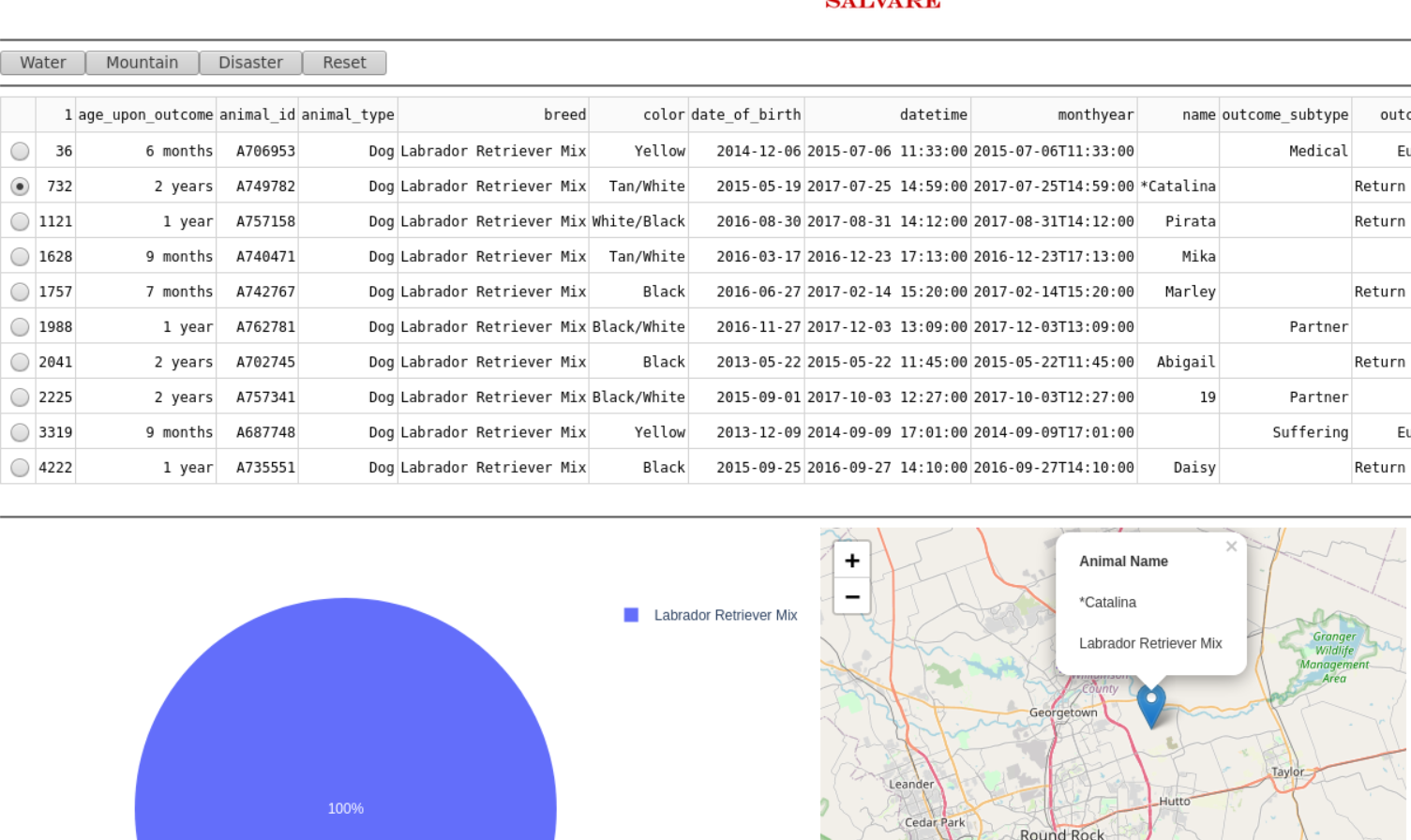
**Chart

Description automatically generated with medium confidence**

**There are four buttons to filter the data.**

**Water**

**The water button will show animals that are suitable for water rescue.**

****

**Mountain**

**The mountain button will filter the data for animals that are suitable for mountain and wilderness rescue.**

**Graphical user interface, chart

Description automatically generated**

**Disaster**

**The disaster button will filter the data for animals that are suitable for disaster search and rescue.**

**Graphical user interface

Description automatically generated with medium confidence**

**Reset**

**The reset button will remove the filter and return all records.**

## Contact

Cory Remick cory.remick@snhu.edu