# Matt Knutson - Grazioso Salvare Search & Rescue README

## About the Project/Project Title

Grazioso Salvare Search & Rescue - Grazioso Salvare will receive data from five animal shelters with the intent to locate dogs that fit the criteria to join a search and rescue training program.

**Motivation**

This project is designed to add CREATE, READ, UPDATE, & DELETE functionality to the Grazioso Salvare animal shelter database. This database will be used to store information about shelter animals. This data will then be used to locate dogs that have the potential to be trained and deployed as rescue animals.

## Installation

Jupyter Notebook – Install from the Jupyter website.

* Jupyter Notebook is an opensource web application. It provides an interface between the Python code and the MongoDB database that allows developers to create graphs, charts, & visuals using their data.

animalShelter.py File – Download from GitHub.

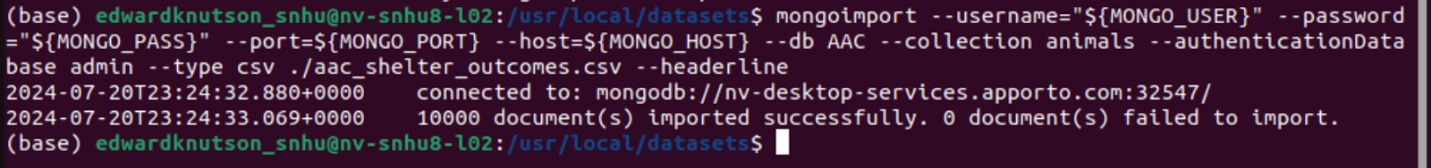
* This project will use Python as a programming language. Python is opensource as well and has an intuitive and easy to learn syntax. It is also a very powerful language that offers a large collection of libraries and resources. Python also provides a large number of libraries that work hand-in-hand with MongoDB software.

The user will need a username & password to access the AAC database and “animals” collection in MongoDB.

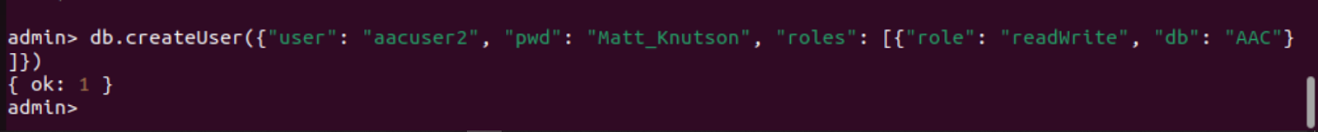
* MongoDB will be used for the database in this project. MongoDB is extremely flexible on how it collects data and provides multiple tools for visualizing your database.

## Getting Started

* The user must import a database, a collection, & a CSV file.



* In MongoDB, create a new user and a password to use with the new database.



* From the Linux shell, login in using the new account.

A computer screen with white text

Description automatically generated

* Then run mongosh to access MongoDB and ensure that the user has the correct permissions, and can access the correct database.

A computer screen shot of a computer code

Description automatically generated

* The user must then create a folder to hold the Python Class animalShelter.py and the Test Script file that will be created with Jupyter Notebook.
* Using Jupyter Notebook, create a testScript file using Python 3 and include it in the folder that holds the animalShelter.py file.
* In the testScript, import the AnimalShelter Class from the animalShelter Python file. EXAMPLE:

from animalShelter import AnimalShelter

* Create a variable that uses the AnimalShelter Class and its functions. Using the AnimalShelter parameters, log in with a username & password. EXAMPLE:

CRUD = AnimalShelter(“username”, “password”)

A screen shot of a computer

Description automatically generated

* Create variables to enable the programs’ internal functions. EXAMPLE:

newEntry = CRUD.create({})

A screenshot of a computer program

Description automatically generated

* Use these variables to CREATE, READ, UPDATE, & DELETE entries from the database.

A close-up of a computer code

Description automatically generated

## Usage

### Code Examples:

**The Python CREATE method.**

**A white background with text

Description automatically generated with medium confidence**

**The Python READ method.**

A screenshot of a computer code

Description automatically generated

**The Python UPDATE method.**

A screenshot of a computer code

Description automatically generated

**The Python DELETE method.**

**A screenshot of a computer code

Description automatically generated**

## Contact

Your name: Matt Knutson