# CS 340 Project One README

## About the Project: CRUD Python Module

*The CRUD Python Module is the Middle-ware layer for a stacked development. It is the “glue” between the base level and the client level.*

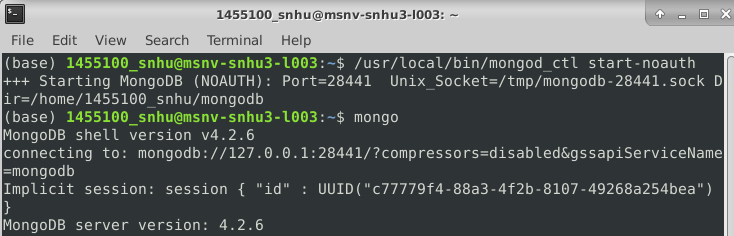
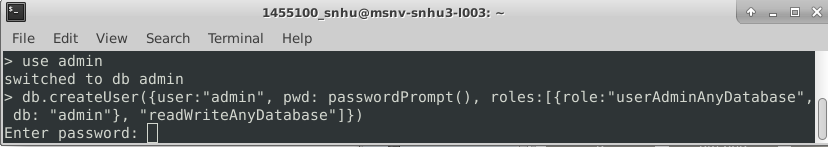
## Motivation

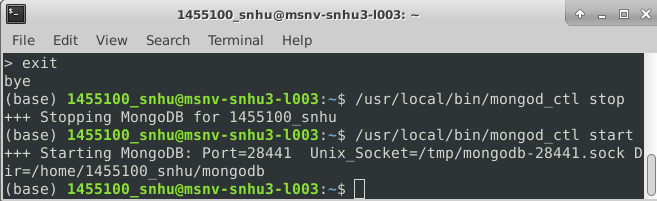
*This module uses PyMongo with MongoDB to provide basic functionality for creating, reading, updating, and deleting documents in the* ***animals*** *collection within the* ***AAC*** *database. The code in this module is reusable and intended to be used by client-facing applications to access and query the database.*

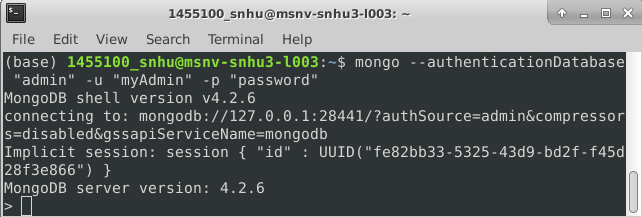
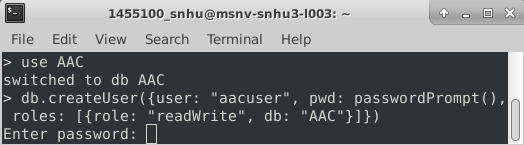
*PyMongo contains the necessary tools for working with MongoDB and is the recommended method to work with MongoDB from Python.*

## Getting Started

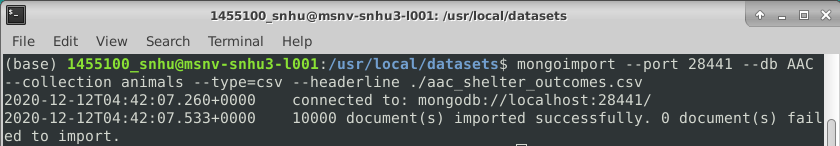
*To get a local copy up and running, follow these simple example steps.*

1. *Install MongoDB, Python, the PyMongo distribution, and project\_one.py   
   (see “Installation” section below)*
2. *Initiate the MongoDB server in your terminal with* ***no authorization*** *by entering the following:****/usr/local/bin/mongod\_ctl start-noauth****Then start mongo by entering* ***mongo****:*
3. *Enter “****use admin****” to enter the admin database, then proceed to enter the “****db.createUser()****” command with the information from the following screenshot. This will create an admin account, and you will be prompted to enter a password for the account:*
4. *Follow the steps shown in the below screenshot to exit mongo and restart it* ***with authentication***
   * *Enter:* ***exit***
   * *Enter:* ***/usr/local/bin/mongod\_ctl stop***
   * *Enter:* ***/usr/local/bin/mongod\_ctl start***



1. *Start mongo again, but this time use the admin account for authorization:****mongo –authenticationDatabase “admin” -u “username” -p “password”*** *\*\*Replace “username” and “password” with the admin account’s username & password:*
2. *Enter “****use AAC****” to enter the AAC database, then proceed to enter the “****db.createUser()****” command with the information from the following screenshot. This will create a user account, and you will be prompted to enter a password for the account:*
3. *You are now ready to import the AAC Outcomes data set and run the project\_one.py program*

## Installation

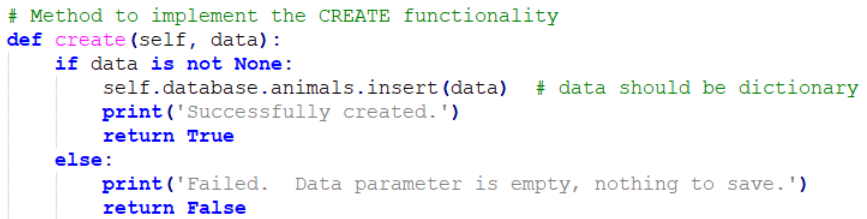
* *MongoDB Community Server:*[*https://www.mongodb.com/try/download/community*](https://www.mongodb.com/try/download/community)
* *Python:*[*https://www.python.org/downloads/*](https://www.python.org/downloads/)
* *PyMongo:*[*https://pypi.org/project/pymongo/*](https://pypi.org/project/pymongo/)
* *After initiating MongoDB in your terminal (see “Getting Started” section above), upload the Austin Animal Center Outcomes data set into MongoDB using the MongoDB csv import tool:*

## Usage

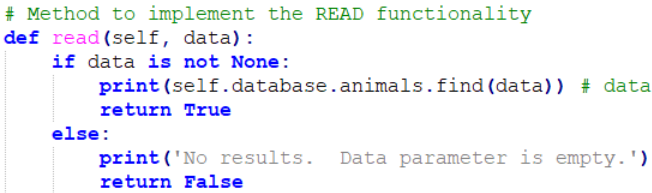
*The main purpose for using the project\_one.py module is for client/web applications to access and query the database on the server side, for creating, reading, updating, and deleting documents. Here are some examples of how the program is setup, how it works, and how to utilize its features.*

### Code Example

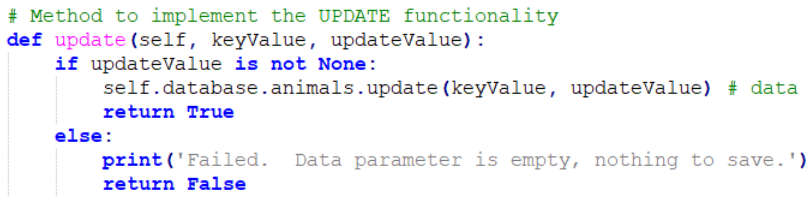
*Below is an example of the* ***create*** *function from project\_one.py. This function allows the client to create a new document in the AAC database and will confirm whether the creation is successful:*

**

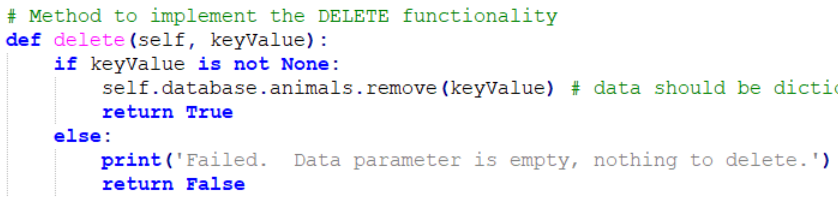
*The* ***read*** *function allows the user to query existing documents in the AAC database by comparing the user’s input data against the documents in the database, then printing the document when it is found. Here is the method that implements the* ***read*** *function:*

**

*The* ***update*** *function can be utilized by the client-side application to locate an existing document and update specified data. Two arguments are required for this function. The first argument is a key/value pair that is used to locate the document in the database, and the second argument is the specific data that is to be updated. Here is an example of the code:*

**

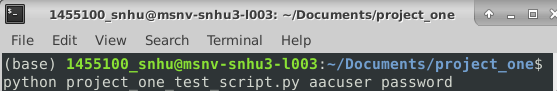
*The* ***delete*** *function is straightforward; it locates a specified document and deletes it from the database. The only argument passed to this function is a key/value pair used to identify the specific document. The code is as follows:*

**

### Tests

*Several sample tests have been established in the script called “project\_one\_test\_script.py”. To run these test samples, download the test script and then open your terminal and enter the following (\*\*the arguments “username” and “password” should be replaced by the credentials for the AAC user account\*\*):*

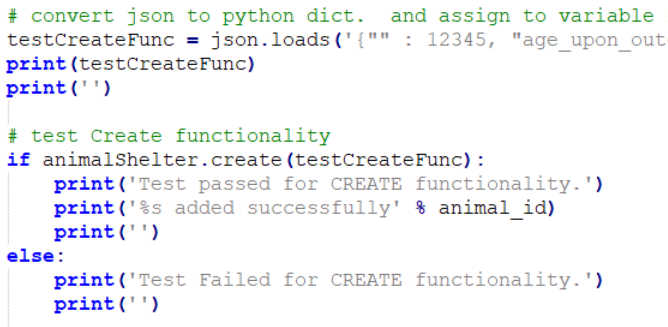
***python project\_one\_test\_script.py username password***



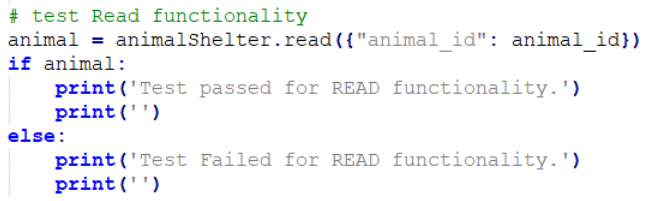
### Screenshots

*The command above will execute the test script, including tests for the* ***create****,* ***read****,* ***update****, and* ***delete*** *functionality. Below are several screenshots of the test script’s code for the* ***create****,* ***read****,* ***update****, and* ***delete*** *function testing, as well as the test results in the terminal once the script is executed.*

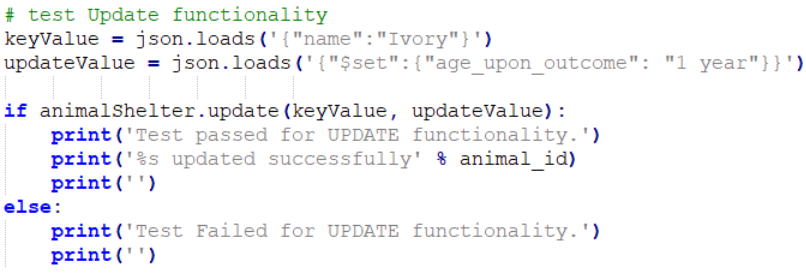
*For the* ***create*** *function test, a set of document data is written as json and converted to a python dictionary then assigned to a test variable. A conditional statement then tests if the function returns True or False, determining whether the function executed properly:*

**

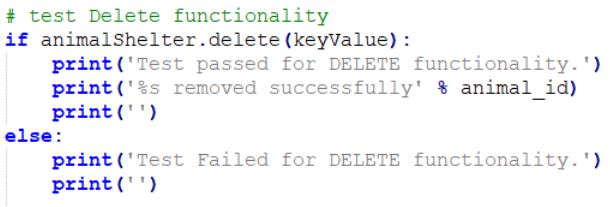
*For the* ***read*** *function test, a key/value pair is used as the argument to locate the document that was created from the previous test.*

**

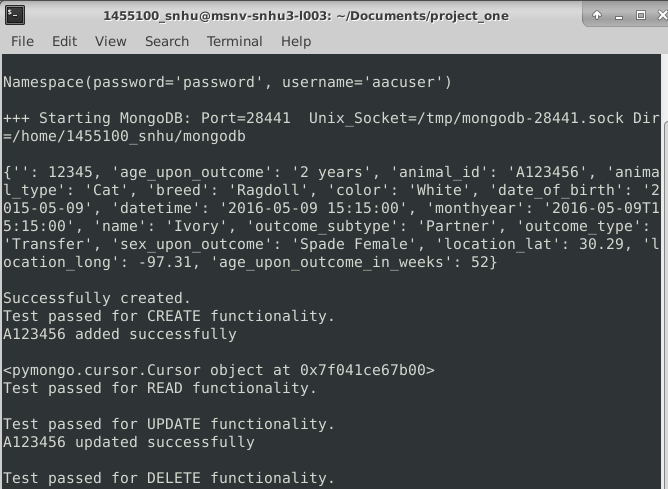
*The* ***update*** *function test sends two variables as arguments to the function. The first argument is a specific key/value pair for the function to use to query the database and locate the corresponding document. The second argument is the data being updated in the selected document:*

**

*The* ***delete*** *function test uses a key/value pair as the only argument, and it is used to locate a specific document which will be delete once it is found:*

**

*Running the project\_one\_test\_script.py in your terminal will yield the following, confirming successful execution of the* ***create****,* ***read****,* ***update****, and* ***delete*** *functions:*



## Roadmap/Features

*I am always looking to improve the functionality of this module. If you have any comments or would like to contribute ideas, please feel free to contact me at the e-mail listed below. Some ideas for future improvements include being able to read and print the entire collection without having to enter any arguments, or updating / deleting multiple documents simultaneously instead of one document at a time.*

## Contact

Brendan O’Connell

[brendan.oconnell2@snhu.edu](mailto:brendan.oconnell2@snhu.edu)