CS-340 5-1: Project One

Keith Breazeale

Southern New Hampshire University

27-March-2023

## About the Project/Project Title

This project is to examine the Create, Read, Update, and Delete functionality in Python with the usage of a Mongo job. The requirements were set forth in Project One, therefore the title of the project is Project One.

## Motivation

This project is to query against the Austin Animal Center Outcome dataset and allow for Create, Read, Update, and Delete queries to be charged against the dataset.

## Getting Started

Make sure you have the csv file containing the data before attempting to set up the dataset.

1. Make sure the Mongo Service is running.
2. Import the dataset.
   1. Text

      Description automatically generated
3. Type in “mongo” from the prompt and verify that the AAC dataset and animals collection is present
   1. Text

      Description automatically generated
4. Create a breed index
   1. Text

      Description automatically generated

## Installation

Python: <https://datascience.com.co/how-to-install-python-2-7-and-3-6-in-windows-10-add-python-path-281e7eae62a?gi=70567cfcaf7d>

Jupyter Notebook: <https://www.dataquest.io/blog/jupyter-notebook-tutorial/>

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

## Usage

### Code Example

Authentication:  
 *def \_\_init\_\_(self,username,password):*

*# init to connect to mongodb without authentication*

*# self.client = MongoClient('mongodb://localhost:43981')*

*# init connect to mongodb with authentication*

*self.client = MongoClient('mongodb://%s:%s@localhost:43981/?authMechanism=DEFAULT&authSource=AAC' % ('aacuser','aacuser'))*

*self.database = self.client['AAC']*

Create Method:

*def create(self, data):*

*if data is not None:*

*self.database.animals.insert(data) # data should be dictionary*

*return True*

*else:*

*raise Exception("Nothing to save, because data parameter is empty")*

*return False*

Read Method:

*#Read Method*

*def read(self, data):*

*if data is not None:*

*return self.database.animals.find\_one(data)*

*else:*

*print('Nothing to read, because data parameter is empty')*

*return False*

Update Method:

*#Update Method*

*def update(self, data, updateData):*

*if data is not None:*

*result = self.database.animals.update\_many(data, {"$set" : updateData})*

*else:*

*return "{}"*

*print ("Record Updated!")*

*return result.raw\_result*

Delete Method:

*#Delete Method*

*def delete(self, data):*

*if data is not None:*

*result = self.database.animals.delete\_many(data)*

*else:*

*return "{}"*

*print("Record Deleted")*

*return result.raw\_result*

### Tests

Using a Jupyter Notebook, execute the below test:

from animal\_shelter import AnimalShelter

shelter = AnimalShelter("aacuser","aacuser")

data = {"1": 3,

"age\_upon\_outcome" : "4 years",

"animal\_id" : "27-Mar-2023\_test",

"animal\_type" : "Dog",

"breed" : "Bluetick Coonhound",

"color" : "Black and White",

"date\_of\_birth" : "2019-01-01",

"datetime" : "2019-01-01 00:00:00",

"monthyear" : "2019-01-01T00:00:00",

"name" : "Courage",

"outcome\_subtype" : "",

"outcome\_type" : "Adoption",

"sex\_upon\_outcome" : "Female",

"location\_lat" : 35.651150,

"location\_long" : -78.746020,

"age\_upon\_outcome\_in\_weeks" : 220}

*#Create Test*

shelter.create(data)

print("Animal added successfully!")

Graphical user interface, text, application

Description automatically generated

*#Read Test*

#Read Test

result = shelter.read({"name" : "Courage"})

print(result)

A picture containing text

Description automatically generated

Update Test

#Update Test

result = shelter.update({"name" : "Courage"}, {"sex\_upon\_outcome" : "Spayed Female"})

print(result)

#Confirm update

result = shelter.read({"name" : "Courage"})

print(result)

A picture containing application

Description automatically generated

Delete Test

#Delete Test

result = shelter.delete({"name" : "Courage"})

print(result)

#Confirm deletion

result = shelter.read({"name" : "Courage"})

print(result)

Graphical user interface, text, application

Description automatically generated

### Screenshots

Graphical user interface, text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

## 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?*

## Contact

Your name: Keith Breazeale