CSCI 3287 Design and Analysis of Data Systems

Project #2 NoSQL Lab --- MongoDB

# Overview

This Project is worth 100 points (out of 1000) toward your final grade. It is due on Tuesday, Nov. 27th, at 11:59 p.m. **No extension can be allowed for this assignment**. Your project submission should be a document saved and submitted as a PDF file via the link found in the Project section of "Week Twelve" in Moodle which is the same place where you found this file.

This project will give you hands-on practice in working with the MongoDB "NoSQL" database software.

# Objectives

1. Students download and install MongoDB
2. Students create a MongoDB database and a collection of documents
3. Students load a lot of meaningful document-based data into the collection
4. Students "query" the document collection to research a topic and answer questions

# Deliverables

Capture screen shots to show evidence of having completed Task # 1 described below. Number each screen shot with the number of the assigned operation/task (1 – 8). Assemble (Copy & Paste) all screen shots into a document. Save the document as a PDF for submission.

# Submission

Use the submission link in the **Project Assignment** section of the Week 12 Moodle -- which is the same place where you got this file.

This is an **individual** assignment, no collaboration allowed. Each student must submit your own final deliverable for this assignment.

# Interview Grading

Interview grading sessions will be scheduled for Wednesday-Thursday-Friday Nov 28-30.

You must attend your interview grading meeting to earn points for Task 2. During the meeting, you will be asked some questions related to this data set. You should bring your laptop computer on which you have installed MongoDB to your interview grading session.

## Introduction:

The following links are helpful for giving an Introduction and basic queries for mongoDB:

1. https://www.tutorialspoint.com/mongodb/mongodb_overview.htm : This link provides a basic overview of mongo db and the basic queries for inserting documents, updating documents, query documents etc.
2. https://www.guru99.com/mongodb-tutorials.html#1 : This is a beginner tutorial and also has links for installing and running mongoDB with images which may be helpful for some students.
3. https://docs.mongodb.com/manual/introduction/ : The official documentation page of mongoDB.

## Installation:

The following links are helpful for finding instructions for the installation of mongoDB:

**Mac System:**

1. https://docs.mongodb.com/manual/tutorial/install-mongodb-on-os-x/ : I have personally tried the installation and it works without issues. One input is that the **.bashrc** file in mac is /Users/apple/.bash_profile.

**Windows System:**

2. https://docs.mongodb.com/tutorials/install-mongodb-on-windows/ : This is the official documentation for windows installation.

**Linux System:**

3. https://docs.mongodb.com/manual/administration/install-on-linux/ : The official instructions for linux installation.
4. https://docs.mongodb.com/manual/installation/ : This link basically shows the details about the operating systems supported by mongoDB.

Students can install the community edition of mongoDB as it is freely available.

In the following 2 tasks, you will have chance to play with different basic operations in MongoDB. Then students can use these tools to learn with a real-world data set, and answer the following questions based on the data set.

### Task 1: Play with different basic operation in MongoDB (20 points)

1. **Create a database:**
    a. use DATABASE_NAME
        i. Ex: use new_mongo_db

ii. The above command will create a database if it does not exist and uses the database if it already exists.
iii. Replace the DATABASE_NAME with the name of the database you want to create.

2. **Drop a database:**
   a. db.dropDatabase()
      i. Ex: >use new_mongo_db
      switched to db new_mongo_db
      >db.dropDatabase()
      ii. First you need to switch to the database that will be dropped. Then use the above command to drop that database.

3. **Creating a collection:**
   a. db.createCollection(name, options)
      i. Ex: >use new_mongo_db
      switched to db new_mongo_db
      >db.createCollection("test_collection")
      { "ok" : 1 }
      ii. The above command creates the collection. But giving some initial options will be highly useful.
         1. >db.createCollection("mycol", { capped : true, autoIndexId : true, size : 6142800, max : 10000 } )
         { "ok" : 1 }
         2. For more information on the options, please check the following link.
            a. https://docs.mongodb.com/manual/reference/method/db.createCollection/
      iii. In mongoDB, it is not necessary to create a collection. When a new document is inserted, mongoDB creates a collection automatically.
4. **Dropping a collection:**
   a. db.COLLECTION_NAME.drop()
      i. Ex: >use new_mongo_db
      switched to db new_mongo_db
      >db.test_collection.drop()
      True
      ii. Firstswitch to the selected database and then use the above command to delete the collection
5. **Insert a document:**
   a. db.COLLECTION_NAME.insert(document)
      i. Ex: db.test_collection.insert({
      _id: ObjectId(7df78ad8902c),

```
  title: 'Mongo Db practice',
  description: 'this class is about MongoDB'
})
```

ii. Replace the COLLECTION_NAME with the name of the collection of your choice

6. **Query a document:**
   a. db.COLLECTION_NAME.find()
      i. Ex: db.test_collection.find().pretty()
      ii. The above query will display the documents present in the collection.

7. **Update a document:**
   a. >db.test_collection.update(SELECTION_CRITERIA, UPDATED_DATA)
      i. Ex: >db.test_collection.update({'Heading':'MongoDB Tutorials'},{$set:{'Heading':'New MongoDB Tutorials'}})
      ii. The above example is used to update the documents that contain 'Heading' as 'MongoDB Tutorials' to 'New MongoDB Tutorials'

8. **Delete Document:**
   a. db.COLLECTION_NAME.remove(DELLETION_CRITTERIA)
      i. Remove only one record:
         1. db.test_collection.remove({ status : "P" },1)
         2. Here the first document which has this key value pair will be deleted.
      ii. Remove all records matching a condition:
         1. db.test_collection.remove({ status : "P" })
         2. Here all the documents which have this key value pair will be deleted.

**Task 2: Use real-world data set to answer the following question (80 Points)**

1. Download the JSON dataset from Moodle, it is named "data.json"

2. From your terminal, type the following command

```
mongoimport --db test --collection population --drop --file ~/desktop/data.json
```

The above command converts the json file and stores it as a set of documents with the collection name of "population".

3.Answer the following sample questions by writing queries and displaying the results. Note: The zip codes will be in the "_id" field of each document in the collection.

- List the five least populated zip codes in Colorado
- List the zip code and population for three zip codes in Brooklyn.
- How many zip codes are there altogether in California, Arizona, and New Mexico?
- List the total population in each state

You do not need to submit anything for Task 2. Your grade for this part will be determined through interview grading with the graders or the professors.

**Expectations in interview grading**

Interview grading sessions will be scheduled for Wednesday-Thursday-Friday Nov 28-30

In the interview grading session, you must attend your grading meeting to qualify for points for Task 2. If you miss your meeting with graders/professor (without notifying graders/professor ahead of time with a suitable reason), this may result in a zero grade for Task 2 of the assignment. The graders/professor are under no obligation to reschedule your appointment if you miss your meeting, so don't miss this !

During the meeting, you will be asked 3-4 questions related to this data set. You will need to type/write the query in your computer/paper and show your output to graders/professors in order to get credit.

You should bring your laptop computer on which you have installed MongoDB to your interview grading session.