Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Lab 7

Database Systems October 29, 2018

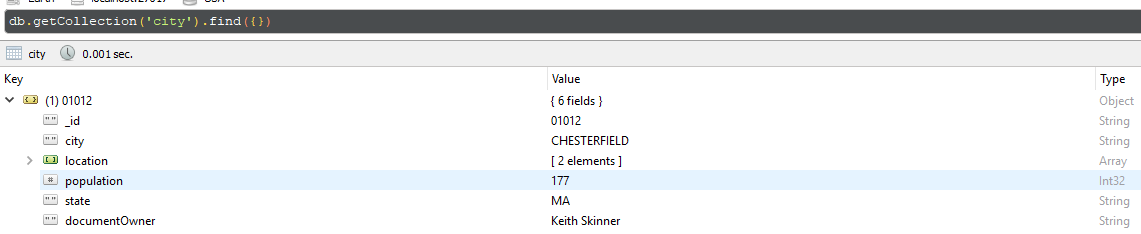
**Part I. Get Ready (10pts)**

1. In your own words, describe the NoSQL approach and how it differs from the traditional database approach. What are the benefits and caveats to each? And when would one approach be preferred over the other? To support your answer, include one relevant example for both the NoSQL database and an RDBMS.  
   \*Complete answers should be around 1/2 page.  
   \*\*Don’t just copy paste from the slides, Wikipedia or online sources. If you use external sources cite them properly.  
   \*\*\*Use one unique and relevant example to support your claims.

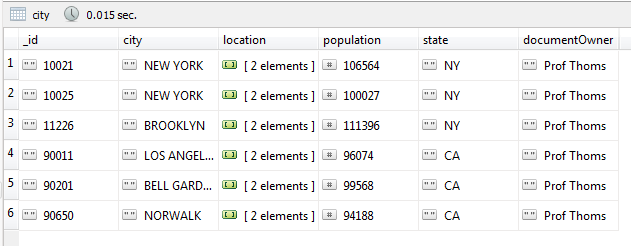
**Part II. Get Set (10pts)**

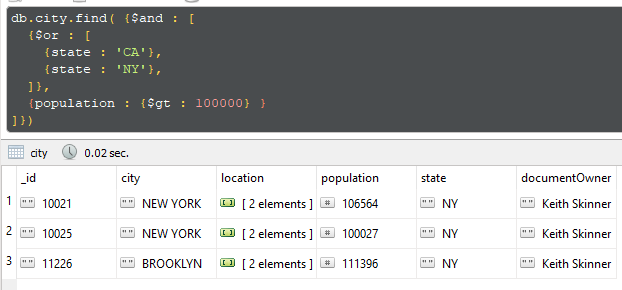
1. Create and run the query to import cities.json into a new database USA under the new collection, city. Update all documents to add the field documentOwner with the value *yourname.*



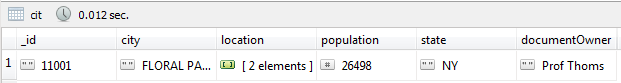
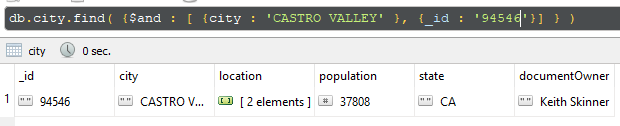


1. Create and run the query to return all documents with information related specifically to California, New York with populations greater than 100k.

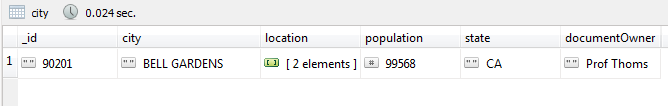


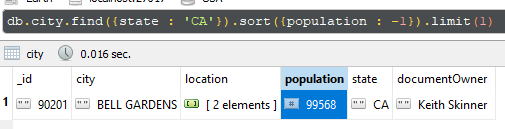


1. Create and run the query to return the document for the city where you grew up.

1. Create and run the query to return the document with the largest population in California.



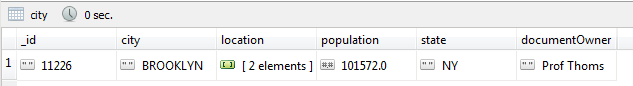


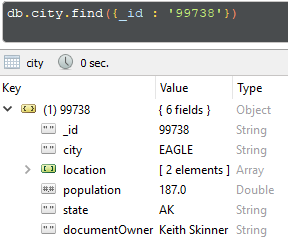
1. Use https://zipwho.com to find the correct population for one new city. Update the document for that city.

Before:



After:





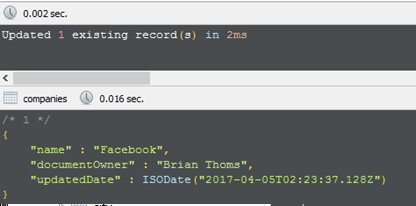
**Part III. Go (10pts)**

Within the USA database, create a new collection named, state, which will store data specific to states. This collection will be used to complete Q8 through Q10.

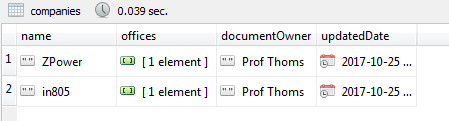
1. Write the insert statements to add documents for cities where you have lived, visited, or wish to visit in the future (up to 5). Include the dates you were there or when you hope to visit. If these documents already exist, update the documents to include the dates you last visited.
2. Write the insert statements to add documents for 5 states. State data is up to you, but include at least three to five pieces of state-related data. Also include a field with your name as the governor.
3. Write the update statements for 5 documents in the city collection based on the information found in the documents created in Q8 (**embed all state data**).
4. Write the update statements for five more documents (not updated in Q9) in the city collection based on the information found in the documents created in Q8 (**reference state documents**).
5. Create the Mongo query that will return state information based on Q10. What issues do you run into when querying MongoDB for referenced data? How can these issues be overcome? Provide 1 example.

**Part IV. Go… (10pts)**

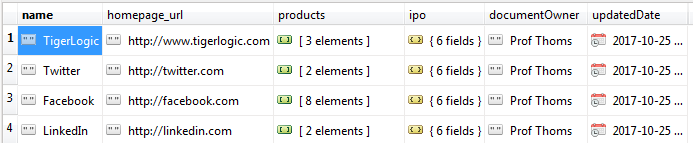
For Part IV, import companies.json into a new database nasdaq, under the collection, companies. For each query, ensure that your name and date is populated as shown below. Do not include the \_id field for any results. The below example returns documents where the company name is Facebook will show two additional fields for documentOwner and updatedDate.



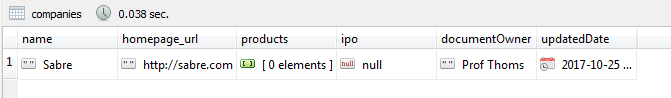
1. Create and run the query to return all companies that have offices in Camarillo, CA. Include fields for the company name, category and address.



1. Create and run the query to return all documents for social-based companies. To determine social companies, use the category code. Include only companies that are publicly traded (IPO is not NULL) and sort your results by the year they were founded in descending order.



1. Create and run the query to return the document for the oldest company where search is a core business function. To determine a company’s core business function, use the tag list.



*Use the mongodb function aggregate() to perform in-line query calculations for the following queries.*

1. Create and run the query to return the company with the most offices in New York.



1. Create and run the query to return the average number of years until IPO for companies that had an initial public offering.

