Assignment 1 – MongoDB

Jing Hua Ye, CIT 03/03/2019

1 MongoDB (CRUD and Simple Queries)

The MongoDB database 'hokey' represents some Hokey team information. We want our database to contain a single collection 'player', with some information of team players. The file 'penguins_player.json' contains an array of 25 JSON objects. Although MongoDB supports schema-less, in this case, the 25 documents of the collection 'player' have the same fields (and each field has the same datatype across the different documents). The JSON object representation of each document (player member) is as follows:

```
{ "position": String,
  "id": integer,
  "weight": integer,
  "height": String,
  "imageUrl": String,
  "birthplace": String,
  "age": integer,
  "name": String,
  "birthday": String,
  "number": integer
}
```

The number represents the number of goals the player has received/shot so far.

Download the penguins_player.json file from the blackboard.

Perform the following steps and answer some question by taking a screenshot for each move you have made:

1. Create a new database hockey. Create a new collection players and list out all collections in your database

2. Insert the following player to your collection

```
{ "position": "Right Wing",
 "id": 8465166,
 "weight": 200,
 "height": "6' 0\"",
 "imageUrl": "http://1.cdn.nhle.com/photos/mugs/8465166.jpg",
 "birthplace": "Seria, BRN",
 "age": 37,
```

```
  "name": "Craig Adams",
  "birthday": "April 26, 1977",
  "number": 27
 }
```

3. Insert the following three players to your collection

```
 { "position": "Right Wing",
  "id": 8465166,
  "weight": 200,
  "height": "6' 0\"",
  "imageUrl": "http://1.cdn.nhle.com/photos/mugs/8465166.jpg",
  "birthplace": "Seria, BRN",
  "age": 37,
  "name": "Craig Adams",
  "birthday": "April 26, 1977",
  "number": 27
},
{ "position": "Right Wing",
  "id": 8475761,
  "weight": 195,
  "height": "6' 2\"",
  "imageUrl": "http://1.cdn.nhle.com/photos/mugs/8475761.jpg",
  "birthplace": "Gardena, CA, USA",
  "age": 23,
  "name": "Beau Bennett",
  "birthday": "November 27, 1991",
  "number": 19
},
{ "position": "Left Wing",
  "id": 8471260,
  "weight": 202,
  "height": "6' 1\"",
  "imageUrl": "http://3.cdn.nhle.com/photos/mugs/8471260.jpg",
  "birthplace": "Meadow Lake, SK, CAN",
  "age": 29,
  "name": "Blake Comeau",
  "birthday": "February 18, 1986",
  "number": 17
}
```

Explain why the profile of the poor player Craig Adams has been inserted twice? How to fix this problem?

4. What happens if we try to insert the profile of the poor player Craig Adams with the same ID?
5. Remove one of the profiles for the player Craig Adams from the collection. Remove all profiles for the player Craig Adams from the collection. Delete all profiles in the players collection.
6. Re-insert the entire team again. The player Craig Adams wants to update his record of the number of goals he has shot from original 27 to 9 in his profile. Update his record of the number of goals he has shot from original 27 to 9 in his profile.
7. The player Craig Adam wants to store list of his favorite cities and movies into his profile. His favorite cities are Chicago and Cheyenne, his favorite movies are "Casablanca", "For a Few Dollars More", and "The Sting". Store list of his favorite cities and movies into his profile.
8. The player Craig Adam wants to add "The Maltese Falcon" movie into his list of favorite movies in his profile. Add "The Maltese Falcon" movie into his list of favorite movies in his profile
9. The player Craig Adam does not want to expose his image in his profile. He decides to remove the entire imageurl from his profile. Remove the entire imageurl from his profile.
10. Find all the defenders of 21 years old

2 MongoDB for Restaurants (Queries and Indexing)

Download the restaurants_dataset.json file from the blackboard. Each document represents 1 restaurant, and provides the following information:

- The name of the restaurant (together with a unique identifier)
- The borough of the city in which the restaurant is placed, together with its complete address (including its zipcode, street, building and coordinates).
- The kind of cuisine offered in the restaurant.
- A list of review from customers, including the date, grade and score of each view.

The following JSON object (used as an example) is one of the documents of the collection:


```
{ "address": {
     "building": "1007",
     "coord": [ -73.856077, 40.848447 ],
     "street": "Morris Park Ave",
     "zipcode": "10462"
  },
  "borough": "Bronx",
  "cuisine": "Bakery",
```

```
  "grades": [
     { "date": { "\$date": 1393804800000 }, "grade": "A", "score": 2 },
     { "date": { "\$date": 1378857600000 }, "grade": "A", "score": 6 },
     { "date": { "\$date": 1358985600000 }, "grade": "A", "score": 10 },
     { "date": { "\$date": 1322006400000 }, "grade": "A", "score": 9 },
     { "date": { "\$date": 1299715200000 }, "grade": "B", "score": 14 }
  ],
  "name": "Morris Park Bake Shop",
  "restaurant\_id": "30075445"
}
```


In the system shell or command prompt, use mongoimport to insert the documents into the restaurants collection in the restaurantdb database. If the collection already exists in the test database, the operation will drop the restaurants collection first.

```
For Linux terminal:
mongoimport --db restaurantdb --collection restaurants --drop --file ~/downloads/restaurants_dataset.json
For Windows cmd:
mongoimport --jsonArray --db restaurantdb --collection restaurants --drop --file
~/downloads/restaurants_dataset.json
```

The mongoimport connects to a mongod instance running on localhost on port number 27017. The –file option provides the path to the data to import, in this case, /downloads/restaurants_dataset.json.

To import data into a mongod instance running on a different host or port, specify the hostname or port by including the –host and the –port options in your mongoimport command.

1. Display the fields restaurant_id, name, borough and cuisine, but exclude the field _id for the first 5 restaurants in Bronx.
2. Find the restaurants who achieved a score more than 90.
3. Find the restaurants which do not prepare any cuisine of 'American ' and achieved a grade point 'A' not belongs to the borough Brooklyn. The document must be displayed according to the cuisine in descending order.
4. Find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Wil' as first three letters for its name.
5. Find the restaurant Id, name, borough and cuisine for those restaurants which contain 'ces' as last three letters for its name.
6. Find the restaurant Id, name, borough and cuisine for those restaurants which contain 'Reg' as three letters somewhere in its name.
7. Find the restaurants which belong to the borough Bronx and prepared either American or Chinese dish

8. Display the name of the cuisine in ascending order and for that same cuisine borough should be in descending order.

9. Whether all the addresses contains the street or not.

10. Evaluate the performance of the following query:
    Find the restaurant Id, name, borough and cuisine for those restaurants which prepared dish except 'American' and 'Chinees' or restaurant's name begins with letter 'Wil'
    Add an index on the cuisine field and an index on name field to see if the performance of the same query will be improved.

3 Report

Produce a two page report on MySQL and MongoDB as methods of storing data, based on your experience in the assignment, the theory covered in lectures and other research. The report should be no more than 2 A4 pages. The report should detail both MongoDB and MySQL advantages and disadvantages as well as examples of where each application would be a good choice for persisting data.

Breakdown of marks for report:

- 10% overview of MongoDB and MySQL
- 10% advantages of each of MongoDB and MySQL
- 10% disadvantages of each of MongoDB and MySQL
- 20% Use your assignment 1 and assignment 2 as your case studies to discuss what kind of applications would be a good choice for using MongoDB and what kind of applications would suit for MySQL.