

MongoDB Lab

The objective of this lab is to familiarize you with MongoDB and its shell. In the following, you will load a data set and write the queries allowing each one to respond to a specific information need.

Introduction

MongoDB is a document-oriented NoSQL database. A MongoDB database consists of a set of collections, themselves made up of documents, built according to a dynamic and not predefined scheme.

A document is an ordered set of key-value pairs, where the keys are strings and the values can be an instance of any predefined data type (null, boolean, number, string, date, regular expression, array, object id, binary data and code), or another document.

Preparation

1. Download, install and launch a MongoDB server.

- www.mongodb.com/download-center#community
- <https://docs.mongodb.com/manual/installation/#mongodb-community-edition>

2. Download the following files:

- [movielens_movies.json](#)
- [movielens_users.json](#)

3. Import these two files into MongoDB:

```
# $MONGO_HOME/bin/mongoimport --db MovieLens --collection movies --file
movielens_movies.json

connected to: localhost

imported 3883 objects

# $MONGO_HOME/bin/mongoimport --db MovieLens --collection users --file
movielens_users.json

connected to: localhost

[...]

imported 6040 objects
```

These commands are used to create the movies and users collections in the MovieLens database and populate them with data from json files.

4. Connect to the MovieLens database:

```
# $MONGO_HOME/bin/mongo MovieLens
MongoDB shell version: 3.2.10
connecting to: MovieLens
>
```

Shell MongoDB

This last command launches the MongoDB shell which is in fact a complete JavaScript interpreter.

So, you can run any JavaScript code there. The shell also offers a certain special command. For example:

- **use <db name>:** used to connect the session to the db name database.
- **show collections:** displays the collections of the current database
- **help:** provides an overview of the most important commands and their use.

Database schema

As said above, the documents in a collection are not subject to a fixed pattern. However, the documents in each collection have a similar structure. We give in this part an example of each collection.

The movies collection contains information about movies, that is, their id, title and genre.

```
> db.movies.findOne()
{
  "_id" : 1,
  "title" : "Toy Story (1995)",
  "genres" : "Animation|Children's|Comedy"
}
```

The users collection contains information about users and their ratings of films. Among the information about the users are their id, name, age, occupation, and gender. The ratings given by each user are represented in a document table, each document containing the id of a movie (referring to the id of the movies collection), the rating assigned and the date on which the user left the rating.

```
> db.users.findOne({}, {movies : {$slice : 2}});
{
  "_id" : 6038,
  "name" : "Yaeko Hassan",
  "gender" : "F",
  "age" : 95,
  "occupation" : "academic/educator",
  "movies" : [
    {
      "movieid" : 1419,
      "rating" : 4,
      "timestamp" : 956714815
    },
    {
      "movieid" : 920,
      "rating" : 3,
      "timestamp" : 956706827
    }
  ]
}
```

Simple queries

Question 1. How many users are there in the database?

- <https://docs.mongodb.com/manual/reference/command/count/>

Question 2. How many movies are there in the database?

Question 3. What is Clifford Johnathan's occupation? Write a request whose response shows only its name and occupation.

- <http://docs.mongodb.org/manual/reference/method/db.collection.find/>

Question 4. How many users are between 18 and 30 years old (inclusive)?

Question 5. How many users are artists (artist) or scientists (scientist)?

Question 6. Who are the ten oldest female authors (writers)?

Question 7. What are all the occupations present in the database?

Insertions, updates and deletions

Question 8. Insert a new user in the database (you, for example).

Question 9. Delete the entry from the database.

Question 10. For all users whose occupation is "programer", change this occupation to "developer".

Question 11. The genres of the movie "Cinderella" should be Animation, Children's and Musical. Modify the corresponding document in a single query so that it contains these three genres without duplicate.

Regular expressions

- <http://docs.mongodb.org/manual/reference/operator/query/regex/>

Question 12. How many films were released in the eighties? (the year of release is indicated in parentheses at the end of the title of each film)

Question 13. How many horror movies are there?

Question 14. How many films are both "Musical" and "Romance" type?

Queries on arrays

Question 15. How many users have rated the movie id 1196 (Star Wars: Episode V - The Empire Strikes Back (1980))?

Question 16. How many users have rated all the films of the first Star Wars trilogy (id 260, 1196, 1210)?

- <http://docs.mongodb.org/manual/reference/operator/query/all/>

Question 17. How many users have rated exactly 48 films?

- <http://docs.mongodb.org/manual/reference/operator/query/size/>

Question 18. For each user, create a num_ratings field which indicates the number of films which he noted.

Question 19. How many users have rated more than 90 films?

Question 20. What are the last three films rated by Jayson Brad?

Aggregation

Question 21. Show how many films were produced in each year of the 1990s; order the results from the most successful year to the least successful.

- <http://docs.mongodb.org/manual/core/aggregation-pipeline>
- <http://docs.mongodb.org/manual/reference/operator/aggregation-pipeline>

Question 22. What is the average rating of the film Pulp Fiction, which has the id 296?

Question 23. In a single query, return for each user their id, name, maximum, minimum and average scores they have given, and order the result by increasing average score.

Question 24. What is the most popular genre in terms of number of notes?

Question 25. What is the highest rated genre (the one with the highest average of all scores)?