Project 4 CS585/DS 503: Big Data Management Spring 2019

"Working with NoSQL Data Engine"

Total Points: 100 points
Given Out: Monday, 1st April, 2019
Due Date: Saturday, 13th April, 2019 (5:00PM)

Teams: Project is to be done in teams, called Project4-team.
Team members will be assigned by CS585/DS503 staff.

Project Scope and Submission

In this project, you will work with one representative NoSQL data engine, in particular, MongoDB. The task is to create, update and query unstructured data using MongoDB. Please see the lecture notes as well as on-line resources such as the MongoDB manual. If you find a resource that you like, please also share it with all students on CANVAS. The MongoDB has been provided to you in the Virtual Machine and/or you can install it also locally on your own machine.

Submission Mechanism

You will submit below as one **single zip file** via CANVAS, namely:

1. This should contain a single text file containing your **MongoDB statements**.
2. This should also include a brief **report (.pdf or .doc)** containing explanations of your solutions and choices and/or issues with using MongoDB.
3. In this report, you should describe **your team's methodology,** i.e., how often you meet, how you communicated, how you shared the work and how finally you derived to an agreed-up project deliverable. **Relative tasks accomplished** by each of your team members are to be specified if the work was not done in close collaboration.
4. **Each team member independently independently** submits peer team comments to CS585/DS503 staff via https://goo.gl/forms/ip0Yw7YbSDG5sXAe2. This is your personal assessment of the team's joint teamwork, your own contributions and effort as well as the contributions and effort of your team member to this project. *These comments will be treated confidentially. You will not be given a grade, until you submit your survey. You are also invited to talk to the CS585/DS503 about your team and/or project, as needed*

Expected Teamwork: *It is expected that all team members would first produce the full solutions for ALL problems to the best of their ability and in a timely fashion. That then as a team you would discuss your solutions, agree on the overall best answer to each problem, and then you submit a final result that everyone in the team agrees to.*

Problem 1: Unstructured Data Modeling and Database Updating. [50 points]

First, you create a MongoDB database, a collection “famous-people”, and insert into this collection the 10 documents from this link: https://docs.mongodb.com/manual/reference/bios-example-collection/ Then apply the following update operations to this collection.

1) Write an operation(s) that changes the _id of “John McCarthy” to value 100.

2) Write an operation(s) that inserts the following new records into the collection:

```
{
   "_id" : 20,
   "name" : {
      "first" : "Mary",
      "last" : "Sally"
   },
   "birth" : ISODate("1933-08-27T04:00:00Z"),
   "death" : ISODate("1984-11-07T04:00:00Z"),
   "contribs" : [
      "C++",
      "Simula"
   ],
   "awards" : [
      {
         "award" : "WPI Award",
         "year" : 1999,
         "by" : "WPI"
      }
   ]
}
```

```
{
   "_id" : 30,
   "name" : {
      "first" : "Ming",
      "last" : "Zhang"
   },
   "birth" : ISODate("1911-04-12T04:00:00Z"),
   "death" : ISODate("2000-11-07T04:00:00Z"),
   "contribs" : [
      "C++",
      "FP",
      "Python",
   ],
   "awards" : [
      {
         "award" : "WPI Award",
         "year" : 1960,
         "by" : "WPI"
      },
      {
         "award" : "Turing Award",
         "year" : 1960,
         "by" : "ACM"
      }
   ]
}
```

3) Report all documents of people who got a “Turing Award” after 1960.

4) Report all people who got more than 2 awards.

5) Update the document of “Guido van Rossum” to add “Python” to the contribution list.

6) Insert a new field called “comments” of type array into document of “Mary Sally” storing the comments: “taught in 3 universities”, “was an amazing pioneer”, “lived in Worcester.”

7) For each contribution by “Mary Sally”, say contribution “C”, list the people’s first and last names who have the same contribution “C”. For example, since “Mary Sally” has two contributions in “C++” and “Simula”, the output for her should be similar to:

{Contribution: “C++”,
People: [{first: “Mary”, last: “Sally”}, {first: “Ming”, last: “Zhang”}]},
{ Contribution: “Simula”,
… .}

8) Report all documents where the first name matches the regular expression “Jo*”, where “*” means any number of characters. Report the documents sorted by the last name.

9) Update the award of document _id =30, which is given by WPI, and set the year to 1965.

10) Add (copy) all the contributions of document _id = 3 to that of document _id = 30.