Assignment 3 – Querying MongoDB

CIS8045 Unstructured Data Management · MM2 · AY2023/2024 Spring

In this assignment, you focus on writing queries to read, update, and aggregate data in MongoDB.

The dataset **kickstarer.json** is given. The data was collected from a crowdfunding website, Kickstarter (www.kickstarter.com). The dataset is about a sample of entrepreneurial fundraising campaigns on Kickstarter. The sample contains around 50K project information. Each project in the data is represented as a JSON document, including fundraising details, category, location, and creator of the project.

Please create a MongoDB script on your own and complete all your works there. First of all, you need to import the data into MongoDB under the database *kickstarter* and collection *projects*. Then write queries to read and analyze data in the database. Once you get the output of queries, take screenshots of your output (similar to the query output on lecture slides) and include them in a separate word document.

You will complete two sets of tasks in this assignment.

Task 1. MongoDB CRUD

In the first task, use MongoDB CRUD operations to answer the questions below. For each question, write queries to get the answers or retrieve the documents:

1.1 Get number of successful projects in "Video Games" category (state is "successful").
1.2 Get the total number of projects in "Video Games" or "Playing Cards" categories.
1.3 Blockbuster projects are those with extremely high pledged amount and backer count. Find the number of blockbuster projects in the data by querying projects with pledged >=$1,000,000 and backers >=10,000.
1.4 Find the top three pledged projects in "Video Games" category; display the output with "_id" and pledged only.
1.5 Update the collection "projects" with a new field "success"; this new field equals to 1, if state is "successful", and 0, otherwise.

Task 2. MongoDB Aggregation Pipeline

In the second task, use MongoDB aggregation pipeline to answer the questions below. For each question, write queries to get the answers or retrieve the documents:

2.1 Get the average pledged amount by project category. Sort your output descending by average pledged amount and limit your output to the first five documents.
2.2 Get the success rate by project category (Hint: use $divide in $project). Limit your output to the first five documents.

2.3 Get the number of projects by each state in US. Sort the output descending by number of projects and limit your output to the first five documents.
2.4 Get the success rate each state in US. Limit your output to the first five documents.
2.5 Sample 10,000 projects, then obtain the creators (creator ID) with at least 3 successful projects.

DELIVERABLE

1. A query script file containing the completed query code.
   - Given the random sampling process in this assignment, the probability is extremely low that two students will get the exact same documents. If two query outputs are the same, we assume that both students did not do independent work.
   - If helpful, you can add necessary comments to explain your queries.
2. A Word document containing your output of queries.

Submit your notebook in iCollege before **April 16, 12:00 pm**. Name your query script as **{LastName}_{FirstName}.js** and word document as **{LastName}_{FirstName}.docx**

Please note that the submission deadline for Assignment 3 is strict. Late submissions will receive 0 points. The solution for the assignment will be shared on April 16 afternoon in iCollege, giving you ample time to review it before the April 17 exam.

Please refer to the course syllabus for the late submission policy, plagiarism policy, and AI-aided tool policy.