Lab 8 Activity - MongoDB

Software System Development – Monsoon 2024

Due Date: 11 October 2024, 05:00 pm

Instructions:

- ➢ Deadline mentioned during the Lab is strictly immutable. No extensions will be given.
- ➢ Any naming convention mentioned in the lab activity must be followed strictly or marks may be deducted for the same.
- ➢ Any plagiarized content will fetch zero marks for the current lab and will be followed by strict action against the students involved. However, discussion of ideas is allowed.

Submission Criteria:

- ➢ Create a folder with your roll number as its name and containing the following files corresponding to the questions:
  - ○ <roll_number>_q1.pdf
  - ○ <README.md
- ➢ Compress the folder as a zip file (name should be <roll_number>.zip) and then upload it on the Moodle before deadline.
- ➢ README.md should contain steps for execution of your script and any extra information like assumptions that you want the evaluator to know before running your script, or any dependencies on some external tools or libraries.
- ➢ For Example:

2023202010.zip

|___2023202010

|___2023202010_q1.pdf

|___2023202010_q2.pdf

|___so on …

|___README.md

(30 Marks : - Q1 to Q6 = 1M, Q7 to Q14 = 3M)

You're given a database with following tables.

Documents, books, students, sales

{refer scripts.pdf file}

You need to write commands for the following queries (Paste the screenshot of your output in the respective pdf file along with the code for the query) : -

1. Using the **sales** collection, find the total revenue generated by each product using an aggregation pipeline.
2. Count the number of books authored by Alice in the **books** collection.
3. Find all distinct product IDs from the **sales** collection.
4. Using the **students** collection, find the average marks scored by each student.
5. Count how many active authors are there in the **books** collection.
6. Find the distinct authors from the **books** collection.
7. Perform a word count on the text field of the **documents** collection.
8. Perform a MapReduce operation to count the number of active books for each author.
9. In the **documents** collection, perform a word count across both the **text** and **title** fields, and count how often each word appears. Group the words based on their presence in either **text** or **title**.
10. In the **books** collection, group the books by **author_name** and **status** (active/inactive) and calculate the total number of books each author has written and the sum of **price** for the books.
11. In the **students** collection, group student marks by **Name** and **Subject**, and calculate the total marks and the number of entries for each subject per student.
12. In the **sales** collection, calculate a 3-day moving average of the **total_sales** for each product. You need to emit sales values by **date** and calculate a rolling average of the last 3 sales days.
13. In the **sales** collection, group sales by **category** and calculate the total revenue, total profit, and the average profit margin for each category.
14. In the **students** collection, compute the average marks for each student and subject. Group the data by student and subject, and return the average marks for each.