NoSQL, CS157C, Summer 2021

HW4 Practice on Document NoSQL Database: Design and implementation of a miniature online Amazon Book Store

Due at 2:29 pm (right before the class time), on July 20, 2021

(0 point for late submission and email submission!)

Notice:

This homework assignment **is for individual work only.** You are not allowed to work as a team. **You are not allowed to use someone's answers or copy from internet**. **Cheating will not be tolerated. You will receive 'F' grade for this course and will be reported to University.** The rule specified in syllabus will be strictly applied. This document is under copyright protected. You are not allowed to upload this document to any websites, otherwise, cheating will be treated (report to SJSU) and your course grade will be changed to F, even after you graduated.

MongoDB is the most popular Document NoSQL Database. This homework is an extension of the hands-on exercise on MongoDB. You will use MongoDB's Document model and JSON to design your database and manipulate data in the database.

Please review the schema carefully listed below. It is a collection of entities used to represent an online Amazon Books Order Management System. Amazon Book Store keeps track of the books in the inventory, the data associated with those books (author, publisher, book details, order) as well as the customers. There are authors, customers, books, and orders. Here are the cardinality relationships:

- A customer can have multiple addresses
- A customer can have multiple orders of books
- A book can have multiple publishers
- Publishers can have multiple addresses
- A book can have many customer's reviews
- A book can belong to multiple categories

The hierarchy structure of entities MUST be followed in your design for the online Amazon Books Order Management System: (30%)

AUTHOR
- FIRST NAME
- LAST NAME
- COUNTRYT

CUSTOMER
- USERNAME

PASSWORD
ACTIVE Y/N (Y = Active account, value of N = Not active account)
ADDRESS
STREET
CITY
ZIP
STATE
COUNTRY
DATE OF CREATION

ORDER
BOOK(S)
ORDER DATE
DELIVERY DATE

BOOK
TITLE
AUTHOR
ISBN
PUBLISHER
NAME
DATE
CITY
ADDRESS
STREET
CITY
ZIP
STATE
COUNTRY
AVAILABLE Y/N (value of Y = Book is in inventory, value of N = sold out)
PAGES
SUMMARY
SUBJECTS
REVIEWS – customers are allowed to leave reviews on the book
CUSTOMER
REVIEW BODY
LANGUAGE

Assume that you are a Software/Data Engineer at Amazon's Online Book Store and you are responsible for the development, implementation and maintenance of the Books Store website using MongoDB NoSQL database with JSON. Your manager assigned the following tasks for you to accomplish:

1. Design a flexible schema/document model for this Books Order Management System using MongoDB. The MongoDB should have COLLECTIONS for AUTHOR, CUSTOMER, ORDER, and BOOK. (10%)
2. Visit the at Amazon's Online Book Store Website and select 5 books. Insert the data instances by your choice into your MongoDB database. Each entity must have at least 5 data instances for simulation/practice purpose. (10%)
3. Write Java with JSON to implement the following functional requirements:
    1) Insert books and users to database. (10%)
    2) Retrieve all information on a specific book (Allow users to enter a book and query all data associated with the book) (10%)
    3) Retrieve all information on the books where the author = "The name of user's interest" (10%)
    4) Retrieve all information on a specific customer where the user id creation is later than 13 July 2017 and the city = "San Jose" (10%)
    5) Retrieve all information on books that have multiple publishers. (10%)
    6) Retrieve all information on the books that have reviews. (10%)
    7) Update a book information. Show the data instance before update and after update. (10%)

1. **Write a professional Report** in MS Word/PDF format with the following items included in the report: **(You will receive 0 point if a report is not submitted.)**
    a) Cover Page – HW4 Title and your name **(-10% if missing)**
    b) Provide screenshots of your model design and data instances stored in the database. **(-30% if missing)**
    c) Organized screenshots and detailed explanations of your code associated with each functional requirement for each query and results. (This is the most important of the HW4. Scores will be added/subtracted based on your work.)
    d) Problems encountered during the implementation: Describe all problems found/encountered during your implementation and how you resolved them. **(-20% if not provided or too simple.)**
    e) Lessons Learned: Describe the concepts and skills you have learned from this Document type NoSQL database exercises and any inspiration or motivation you got

from this exercise vs other NoSQL databases and Relational Database. **(-20% if not provided or too simple.)**

Submission Requirements:

1. Submit **HW4-Last_Name.zip including all source code files and the report in pdf**. Any file with incorrect file name will be taken off 20%.
2. Warning: DO NOT wait until the last minute to upload your file to Canvas. There will be excuse for a late submission.