Assignment 3 (10% of total marks)

Due date: Monday, 27 May 2024, 9:00 pm Singapore time.

Scope:

The tasks of this assignment cover the implementation of hierarchical data structures as BSON documents, query language of MongoDB database systems, aggregations and cursors and implementation of data manipulations on BSON documents.

Assessment criteria:

Marks will be awarded for:

- Correct,
- Comprehensive, and
- Appropriate

application of the materials covered in this subject.

Only electronic submission through Moodle at: https://moodle.uowplatform.edu.au/ is accepted. All email submissions will be deleted and mark 0 ("zero") will be awarded.

For all the implemented tasks, your report or output must include a listing of all JSON scripts and their output.

The submission procedure is explained at the end of this specification.

Assignment Specification:

Preliminary actions

In this implementation task, you may use the virtual machine that runs MongoDB Enterprise Edition database server 3.6.5. or the actual installation of MongoDB in your computer system.

Download and unzip a file Assignment3-all-files.zip. You should get the specification of the Assignment 3 and a Java script file, customerShoppingcartV2.js.

Implement each of the specified tasks and **save the solution for each task together with its results** in a file $solutionX.lst$ (where 'X' is the task number, e.g., 1, 2,…). You can process the queries one by one and later on $copy$ the content of Terminal window and $paste$ it one by one into a file and save it as $solutionX.lst$. The results of query processing must be included into the file $solutionX.lst$ in the same orders as the queries

listed above.

Alternatively, you can create a Java script file (solutionX.js) that contains all your solutions to the questions. In the example shown below, it is the execution of solution1.js for Task 1 (example). When ready, open a Terminal window and process your script at command shell prompt in the following way.

```
mongo –port 4000 databaseName < solution1.js > solution1.lst
```

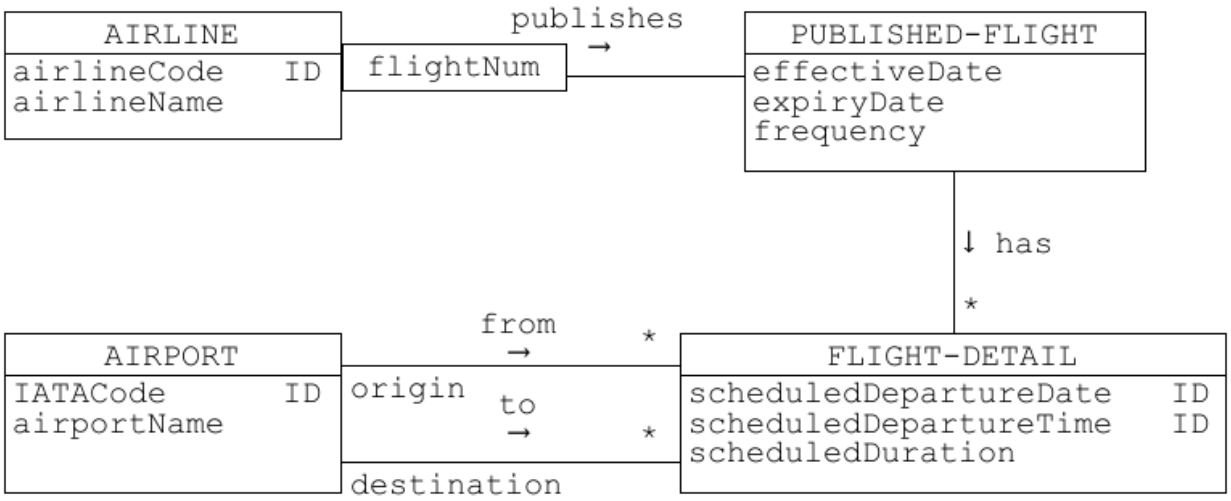
Note, that there is no need to connect to MongoDB command line interface.
'databaseName' is the databaseName where you created the collections.

Task 1 (4.0 marks)
Implementation of BSON documents

Consider the following conceptual schema of a sample database that contains information about a bank, the bank-accounts administered, the owner's bank account, and the transactions made.

Consider the following domain specification of an application and its conceptual schema:

An airline publishes a PublishedFlights on a recurring basis. The flightNumber uniquely identifies each PublishedFlight for an Airline. Each PublishedFlight has one or more FlightDetail, each from some origin Airport to a destination Airport.
An airport has an IATA (International Air Transport Association) code and a name. For example, the name of Singapore's airport is Singapore Changi Airport, and its IATA code is SIN.



(1) Create a new MongoDB database with a name that is the same as your UOW student number.

(2) Create a collection with a name the same as the prefix of your UOW email address, e.g., sjapit.

(3) Insert into the collection the documents whose contents are consistent with the conceptual schema given above. It is important that your documents represent the hierarchical structures in the best possible way. You do not need to enforce all identification constraints determined in the conceptual schema. Insert at least two airlines, two published-flight, three flight-detail, and three airports involving the airlines; each airline publishes two published-flight, and each published-flight has two flight details. You can use the following IATA code and the airport names for your airport entries.

IATA Code	Airport name
SIN	Singapore Changi Airport
HLP	Halim Perdana Kusuma Airport Jakarta
CKG	Chongqing Jiangbei International Airport
KUL	Kuala Lumpur International Airport
DMK	Don Muang Airport
STV	Surat International Airport

The rest of the values for the other attributes are up to you but must be sensible.

(4) List the contents of the collection created in step (2) and loaded with the documents in a step (3) in a pretty format.

Save a report from the processing of actions listed above in a file solution1.lst. To create a "report, you can *copy* the contents of Terminal window and *paste* it into a file solution1.lst. The results of all actions must be included in solution1.lst in the same order as their specifications listed above.

Deliverables

Generate a file solution1.lst with a report from implementation and processing of the actions listed above. The report MUST have no errors. The report must list all methods processed. The report MUST list in a pretty format all documents inserted into a collection. Please save your solution1.lst in pdf format before submission.

Task 2 (12.0 marks)
Aggregation and Data manipulations on BSON documents

If you have not done it yet, start Mongo client and connect to the MongoDB database server. Next, process the script file *customerShoppingcartV2.js* to create the BSON documents into a collection *shoppingCart*. Each of the following questions is worth 1.5 marks. Make yourself familiar with the contents of the collections.

1. List the customer's name, address and the created shopping cart who purchased product P1003.

2. List the customer's name, address, and the created shopping cart of the customer who created a shopping cart on 11 May 2024 (ISODate('2024-05-11T00:00:00Z')). Please do not show the customer's id.

3. Find the total number of shopping carts created by each customer. For each customer, list his/her email address and the total number of shopping cart created.

4. Find the products that have been included in at least 2 or 3 shopping carts.

5. For each price base, list the price base and the total number of each price base.

6. Find the customers who have purchased both the products 'P1002' and 'P1003'.

7. Find the products that have not been included in any of the shopping carts.

8. Find the total number of customers who do not provide his/her telephone number.

9. Update the closing date (dateClosed) of the cart 'cart001' of the customer 'C92378' to 15 May 2024. (Hint. You can use the function new Date("2024-05-15" to set the date.)

10. Delete from the collection a shoppingcart (cart005) created by the customer C78263.

Process each query implemented and **save each query together with its results** in a file *solution1.pdf*. You can process the queries one by one and later on *copy* the content of Terminal window and *paste* it one by one into a file and save the file as *solution1.pdf*. The results of query processing must be included into the file *solution1.pdf* in the same orders as the queries listed above.

Alternatively, you can create a Java script file (solution2.js) that contains all your solutions to the questions. When ready, open a Terminal window and process your script at command shell prompt in the following way.

```
mongo –port 4000 databaseName < solution2.js > solution2.lst
```

Note, that there is no need to connect to MongoDB command line interface.
'databaseName' is the databaseName you created the collections.

Deliverables

Generate a file solution2.lst with a report from processing the queries listed above. The report MUST have no errors. The report must list all methods processed by mongo command line shell.

Submissions

This assignment is due by Monday, 27 May 2024, 9:00 pm Singapore time.

Submit the files **solution1.lst, solution1.js, solution2.lst, solution2.js, solution3.lst and solution3.js** through Moodle in the following way:

1) Zip all the files (Solution1.lst, solution1.js, solution2.lst, solution2.js, solution3.lst, and solution3.js into one zipped folder. Name your zipped file as YourName-A3). **Please do not save your solutions in .pdf format. I need to execute your .js scripts.**
2) Access Moodle at **http://moodle.uowplatform.edu.au/**
3) To login use a Login link located in the right upper corner the Web page or in the middle of the bottom of the Web page
4) When successfully logged in, select a site CSCI235 (SP224) Database Systems
5) Scroll down to a section Submissions of Assignments
6) Click at Submit your Assignment 3 here link.
7) Click at a button Add Submission
8) Move the zipped file created in Step 1 above into an area provided in Moodle. You can drag and drop files here to add them. You can also use a link *Add…*
9) Click at a button Save changes,
10) Click at check box to confirm authorship of a submission,
11) When you are satisfied, remember to click at a button Submit assignment.

A policy regarding late submissions is included in the subject outline.

Only one submission per student is accepted.

Assignment 3 is an individual assignment, and it is expected that all its tasks will be solved individually without any cooperation with the other students. Plagiarism is treated seriously. Students involved will likely receive zero. If you have any doubts, questions, etc. please consult your lecturer or tutor during lab classes or over e-mail.

End of specification