

Assignment 3 (15% of total marks)

Due date: Thursday, 24 November 2022, 11:59 pm (midnight) Singapore time.

Scope:

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

Process each query implemented and **save each query together with its results** in a file *solution1.lst*. 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 *solution1.lst*. The results of query processing must be included into the file *solution1.lst* in the same orders as the queries listed above.

Alternatively, you can create a Java script file (solution1.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.

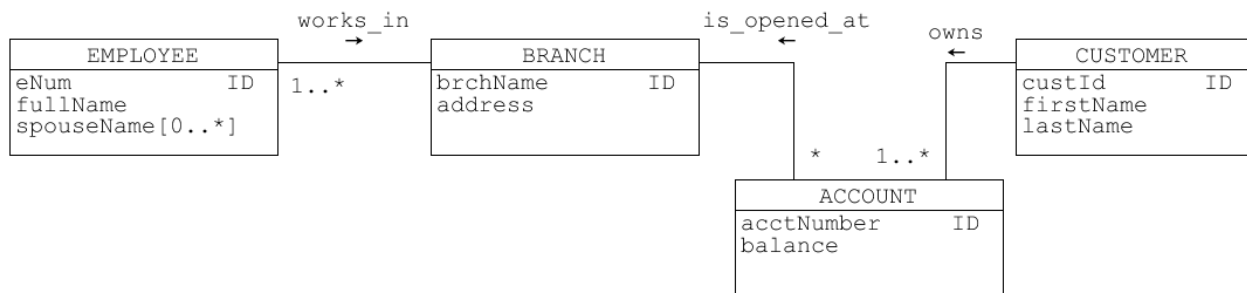
```
mongo -port 4000 databaseName < solution1.js > solution1.lst
```

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

Task 1 (2.0 marks)

Task 1 Implementation of BSON documents

Consider the following conceptual schema of a sample MongoDB database.



Start Mongo client and connect to the MongoDB database server. Next, implement the following actions.

- (1) Create a new MongoDB database with a name the same 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 one document for each class of objects given in the schema with the meaningful values of all properties describing the classes of objects in the schema.
- (4) List the contents of the collection created in a 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 into a file 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 (5.0 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 a script file `empeProject.js` to insert BSON documents into a collection `empeProject`. Make yourself familiar with the contents of the collection.

Use either a method `find()` or a method `aggregate()` available in MongoDB to write the implementations of the following queries. Implementation of each query is worth 1 mark.

- (i) Find the first name (`fName`) and last name (`lName`) of all employee who have experience in Database Design. Do not show the object identifier (`_id`).

(1.0 mark)

- (ii) Find the first name (`fName`), last name (`lName`) and experience (`experience`) of the employee with employee id (`empeId`) = "e002". Display only the employee first name, last name and the experience. Do not display the object identifier (`_id`).

(1.0 mark)

- (iii) Find all employees who possess 4 experiences. Show only the employee's information.

(1.0 mark)

Use the method `update()` to write the implementations of the following data manipulation operations. Implementation of each data manipulation operation is worth 1 mark.

- (iv) Add a new experience "HIVE" to the employee whose `empeId` is 'e001'.

(1.0 mark)

- (v) Change the email account for employee e001 to "jamesbond\$hotmail.com".

(1.0 mark)

Save the listings of processed queries and the results of processing in a file `solution2.lst`. To create a report, you can process the queries one by one and later *copy* the contents of Terminal window and *paste* it into a file `solution2.lst`. The results of query processing must be included into a file `solution2.lst` in the same order the as respective queries listed above.

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.

Task 3 (8.0 marks)

Applying aggregation to search MongoDB documents.

If you have not done yet, start Mongo client and connect to the MongoDB database server. Next, process the script file `subject.js` to insert JSON documents into a collection `Subject`. Each question worth 0.8 marks. Make yourself familiar with the contents of the collections.

- 1) Find the total number of subjects.
- 2) Find the total number of subjects that have no prerequisite.
- 3) Find the total number of subjects that worth more than 3 credit points.
- 4) Find the subject title, subject type, and credit value of subject with the highest credit value.
- 5) Find the subject title, subject type, and credit value of subjects with credit value = 3. List the output in ascending order by subject title.
- 6) For each subject type, list the name of subject type and the total number of subject which belong to the subject type.
- 7) Find the ISBN and title of books that was published in 2022.
- 8) Find the title, author, and type of books used for the subject CSCI235.
- 9) Find the ISBN, title of book and publisher of books that have at least two authors and at most three authors.
- 10) List the subject code, title of book and publisher of books. Sort the results by subject code in ascending order, and the publisher in the descending order.

Process each query implemented and save each query together with its results in a file `solution3.lst`. You can process the queries one by one and later *copy* the content of Terminal window and *paste* it one by one into a file `solution3.lst`. The results of query processing must be included into a file `solution3.lst` in the same orders as the queries listed above.

Submissions

This assignment is due by Thursday, 24 November 2022, 11:59 pm (midnight) Singapore time.

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

- 1) Zip all the files (Solution1.lst, solution2.lst and solution3.lst into one zipped folder. Name your zipped file as YourName-A3)
- 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 (SP422)
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