Getting started with MongoDB

***Weight: 15% of final grade***

Learning outcomes

Upon successful completion of this Assignment, you will have demonstrated the abilities to:

* Create a mongoDB document using key:value pairs.
* Define values of different data types within a mongoDB document.
* Insert a mongoDB document into a collection within the test database.
* Run simple commands to retrieve one document of a collection within the test database.
* Import data from a json file into a collection in MongoDB database.
* Perform ad hoc queries on the database using the find or findOne functions and a query document.
* Query for ranges, set inclusion, inequalities, and more by using $-conditionals.

Group work acknowledgment

We, ------------(mention your names), declare that the attached assignment is our own work in accordance with the Centennial Academic Policy. No part of this assignment has been copied manually or electronically from any other source (including web sites) **or distributed to other students.**

Specify below what each member has done towards the completion of this work:

Name Task(s)

1-

2-

3-

MongoDB installation and use

* Download and install MongoDB community edition.

Follow the instructions in the link below. <https://docs.mongodb.com/manual/tutorial/install-mongodb-on-windows/>

* + Download MongoDB Community Edition installer.
  + Run the MongoDB installer.
  + Follow the MongoDB Community Edition installation wizard
    - Choose Setup Type: complete
    - Service Configuration: Windows service
    - Run the service as Network Service user
  + From Windows Explorer/File Explorer, go to C:\Program Files\MongoDB\Server\4.4\bin\ directory and double-click on [mongo.exe](https://docs.mongodb.com/manual/reference/program/mongo/#bin.mongo).

Instructions

You are to write mongoDB commands in the mongoDB shell to create a new database, populate a collection, and insert a document into the collection. Paste the code and screen shot from the shell for each question solution.

1. Display the database you are using. Type db

The operation should return test, which is the default database.

A picture containing text, screenshot, electronics, display

Description automatically generated

1. Switch to database called A4. Type use A4

Even if it is non-existing, MongoDB will create a database called A4. Type again db

The operation should return A4

A screenshot of a computer

Description automatically generated with medium confidence

1. Use insert() to insert the following item in the products collection.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| code | desc | indate | qoh | min | price | Colors | Size | Package |
| 11QER/31 | Power painter, 15 psi., 3-nozzle | 2011-11-03 | 100 | 5 | 109.99 | Black, red | l: 24  w:19.99  h:14.99  uom: "cm" | weight: 1.79 uom: "kg" |

db.products.insert({code:'11QER/31', desc:'Power painter, 15 psi., 30nozzle',indate:ISODate("2011-11-03"), goh:100, min:5, price:109.99, Colors:'Black,red', Size:['l:24','w:19.99','h:14.99','uom:cm'], Package:['weight:1.79', 'uom:kg']})

Text

Description automatically generated

1. Use findOne() and show the document in the products collection. Use pretty() for formatting.

db.products.findOne({code:"11QER/31"})

A screenshot of a computer screen

Description automatically generated with medium confidence

1. Use insertMany() to bulk insert the following items in the products collection. When the data is missing for a specific attribute, the document should not contain that attribute as key in its structure.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| code | desc | indate | qoh | min | price | Colors | Size | Package |
| 13-Q2/P2 | 7.25-in. pwr. saw blade | 2011-12-13 | 32 | 15 | 14.99 | - | w: 7.25  h: 0.25  uom: in | - |
| 14-Q1/L3 | 9.00-in. pwr. saw blade | 2011-11-13 | 18 | 12 | 17.49 | Metallic | l: 9  uom: in | weight: 0.6  uom: lbs |
| 1546-QQ2 | Hrd. cloth, 1/4-in., 2x50 | 2012-01-15 | 15 | 8 | 39.95 | silver | l:50  h:2  uom: ft | weight: 19.5  uom: lbs |
| 1558-QW1 | Hrd. cloth, 1/2-in., 4x50 | 2012-01-15 00:00:00.000 | 23 | 5 | 43.99 | - | l:50  h:4  uom: ft | weight: 40  uom: lbs |

db.products.insertMany([{code:'13-Q2/P2', desc:'7.25-in. pwr. saw blade',indate:ISODate("2011-12-03"), goh:32, min:15, price:14.99, Size:['w:7.25','h:0.25','uom:in']},{code:'14-Q1/L3', desc:'9.00-in. pwr. saw blade',indate:ISODate("2011-11-13"), goh:18, min:12, price:17.49, Colors:'Metallic', Size:['l:9','uom:in'], Package:['weight:0.6', 'uom:lbs']},{code:'1546-QQ2', desc:'Hrd. cloth, 1/4-in., 2x50',indate:ISODate("2012-01-15"), goh:15, min:8, price:39.95, Colors:'silver', Size:['l:50','h:2','uom:ft'], Package:['weight:19.5', 'uom:lbs']},{code:'1558-QW1', desc:'Hrd. cloth, 1/2-in., 4x50',indate:ISODate("2012-01-15 00:00:00.000"), goh:23, min:5, price:43.99, Size:['l:50','h:4','uom:ft'], Package:['weight:40', 'uom:lbs']}])

Graphical user interface, text

Description automatically generated

1. Show all inserted documents in the products collection. Use find() an pretty() function for formatting.

db.products.find().pretty()

Text

Description automatically generated

1. Remove product 11QER/31 from the products collection. Use the remove function and use code as key. Show all documents in the products collection. Use find() an pretty() function for formatting. Make sure product 11QER/31 does not exist.

db.products.remove({code: '11QER/31'})

Text

Description automatically generated

Text

Description automatically generated

Data:

Restaurant-data.json

Importing data

Download MongoDB database tools zip file. <https://www.mongodb.com/try/download/database-tools?tck=docs_databasetools>

unzip the folder and place it under C:\Program Files\MongoDB. You can rename the folder as mongo-db-database-tools.

Create a folder named Mongo\_Import\_Export under C:\Program Files\MongoDB and place the restaurant-data.json file under mongo\_Import\_Export folder.

The folder C:\Program Files\MongoDB should now contain:

Graphical user interface, text, application

Description automatically generated

In mongodb-database-tools/bin folder, double click on mongoimport, this will open up a command prompt. Or open a command prompt, type cd pathTo

pathTo is the path to where mongoimport is located.

1. It is time to import the data from the json file into a mongodb database. For this, type the following command:

mongoimport.exe "c:\MongoDB\Mongo\_Import\_Export\ restaurant-data.json" /d restaudb /c restaurants /jsonArray

this should show a message that 10 documents are imported in the restaurants collection as follows:



A new mongoDB database restaudb will be created and a new collection named restaurants will be created in restaudb database.

1. Go to mongo terminal, type the following.

use restaudb

show collections

this should show the restaurants collection.

Text

Description automatically generated

db.restaurants.find()

This should show all restaurant documents in the restaurants collection.

querying mongodb

Write MongoDB queries to answer each of the following questions.

1. Show one document of the restaurants collection in A4 database.

db.restaurants.findOne()

Text

Description automatically generated

**Restrict the search**

1. Find the restaurants with name “The Dutch”.

db.restaurants.find({name:'The Dutch'}).pretty()

Text

Description automatically generated

1. Find the restaurants that have cuisine type “American”.

There should be documents 4, 6, 7, 8.

db.restaurants.find({cuisine\_type:'American'}).pretty()

Text

Description automatically generated

1. Show the restaurants that are located in the “Manhattan” neighborhood and have American cuisine type.

db.restaurants.find({cuisine\_type:'American', neighborhood:'Manhattan'}).pretty()

Text

Description automatically generated

**Specifying which keys to return.**

1. Show all restaurants with all their key/value pairs except the reviews and the \_id.

**Query conditionals:** "$lt", "$lte", "$gt", and "$gte", "$ne"

1. Show the restaurants whose id is between 3 and 5 inclusive. Documents 3, 4, 5 should show in the output.

db.restaurants.find({id:{'$in':[3,4,5]}}).pretty()

Text

Description automatically generated

1. Find the restaurants that do not belong to the American cuisine type. Show only the id and cuisine type.

db.restaurants.find({cuisine\_type:{$nin:['American']}},{\_id:0, id:1, cuisine\_type:1}).pretty()

Text

Description automatically generated

**Or queries**

* "$in" can be used to query for a variety of values for a single key.
* "$or" is more general; it can be used to query for any of the given values across multiple keys.
* The opposite of "$in" is "$nin", which returns documents that don’t match any of the criteria in the array.

1. Find the restaurants that have the cuisine type “Pizza” or “Mexican”. Use the "$in" operator. Show only id and cuisine\_type. The output should show document id 2, 5 and 10.

db.restaurants.find({$or:[{cuisine\_type:{$in:['Pizza','Mexican']}}]},{\_id:0, id:1, cuisine\_type:1}).pretty()

Text

Description automatically generated

1. Find the restaurants that belong to "Brooklyn" neighborhood and have cuisine type "Pizza". Show only the id, name, cuisine\_type and neighborhood keys. The output should have only restaurant id 1. ？？？

db.restaurants.find({$and:[{neighborhood:'Brooklyn'},{cuisine\_type:'Pizza'}]},{\_id:0, id:1, cuisine\_type:1, neighborhood: 1}).pretty()

Text

Description automatically generated

**Querying arrays**

$all, $size

The $slice operator

1. Use the $slice operator to show the last review added to restaurant id 1.

**db.restaurants.find({id:1},{reviews:{$slice:-1}}).pretty()**

Text

Description automatically generated

### **Querying on Embedded Documents**

use "$elemMatch" when you have more than one key you want to match on in an embedded document.

1. Find the reviews provided by user “Jason” with rating equal 4 on restaurants whose cuisine\_type is Mexican. Show the id, name, cuisine\_type and only the matched review in the output.

SUBMISSION

Each question is worth 5 pts.

Submit your A4\_GroupX.doc file on dropBox. Replace X with your group number.

If a student does not contribute to the work, do not list his/her name(s) under the group section in the Assignment file and will get 0.