FIT9132 Introduction to Databases

Week 12 Tutorial Activities

Big Data and NO SQL

FIT Database Teaching Team

Complete the week 12 activities listed below

12.1 MongoDB
- 12.1.1 Lead tutor Demo
- 12.1.2 MongoDB Create Update and Delete
- 12.1.3 MongoDB Read

12.2 SETU and Assignment 2

FIT9132 2020 S1

FIT9132 Introduction to Databases

Author: FIT Database Teaching Team

Important

Remember before starting any lab activity which involves working with files, first use SQLDeveloper to pull from the FIT GitLab server so as to ensure your local files and the FIT GitLab server files are in sync.

12.1 MongoDB

12.1.1 Lead tutor Demo

In this tutorial we will focus on MongoDB, one of the popular Big Data platforms. The data in MongoDB must be stored in a specific format, for example:

```
{
    "_id": 11111111,
    "name": "Mary Smith",
    "contactInfo": {
      "address": "20/1 Princess Highway, Caulfield East, VIC , 3145",
      "phone": "0411111222     ",
      "email": "msmith@monash.edu"
    },
    "enrolmentInfo": [
      {
        "unitcode": "FIT1004",
        "year": "2013",
        "semester": 1,
        "mark": 65,
        "grade": "C "
      },
      {
        "unitcode": "FIT1040",
        "year": "2013",
        "semester": 2,
        "mark": 74,
        "grade": "D "
      },
      {
        "unitcode": "FIT1040",
        "year": "2013",
        "semester": 1,
        "mark": 45,
        "grade": "N "
      },
      {
        "unitcode": "FIT2077",
        "year": "2013",
        "semester": 2,
        "mark": 74,
        "grade": "D "
      }
    ]
  }
```

We can generate JSON from relational database data and use the JSON to build up the data in MongoDB. To generate JSON use:

```
SET PAGESIZE 50

SELECT
    JSON_OBJECT ( '_id' VALUE studid, 'name' VALUE studfname
                || ' '
                || studlname,
                'contactInfo' VALUE JSON_OBJECT (
                    'address' VALUE studaddress,
                    'phone'   VALUE studphone,
                    'email' VALUE studemail
                    ),
                'enrolmentInfo' VALUE JSON_ARRAYAGG(
                    JSON_OBJECT(
                        'unitcode' VALUE unitcode,
                        'year' VALUE to_char(ofyear, 'yyyy'),
                        'semester' VALUE semester,
                        'mark' VALUE mark,
                        'grade' VALUE grade
                        )
                    ) FORMAT JSON )
    || ','
FROM
    uni.student
    NATURAL JOIN uni.enrolment
GROUP BY
    studid,
    studfname,
    studlname,
    studaddress,
    studphone,
    studemail;
```

Store the script output as studentenrolment.txt.

To see the structure of the data you could use copy paste the text into the Configuration box at https://mongoplayground.net/. Make sure that you paste the text within the bracket **[paste your text here]** then click format.

Now, create the collection on MongoDB. Open MongoDB shell https://docs.mongodb.com/manual/tutorial/getting-started/ then click the working window to connect to the database.

A MongoDB database is a set of collections, a collection is a set of documents. The default database used in the MongoDB shell is the *test* database.

Let's create a collection and do some document manipulations on MongoDB

1. Create a collection and insert one document into the collection. For this activity, we call the collection as studentenrolment

   ```
   db.studentenrolment.insertOne({"_id":11111111,"name":"Mary
   Smith","contactInfo":{"address":"20/1 Princess Highway, Caulfield
   ```

```
East, VIC , 3145","phone":"0411111222
","email":"msmith@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT100
4","year":"2013","semester":1,"mark":65,"grade":"C
"},{"unitcode":"FIT1040","year":"2013","semester":2,"mark":74,"grad
e":"D
"},{"unitcode":"FIT1040","year":"2013","semester":1,"mark":45,"grad
e":"N
"},{"unitcode":"FIT2077","year":"2013","semester":2,"mark":74,"grad
e":"D "}]},
)
```

2. Insert many documents (9 extra documents) into the collection

```
db.studentenrolment.insertMany([{"_id":11111112,"name":"Matthew
Long","contactInfo":{"address":"20/1 Princess Highway, Caulfield
East, VIC , 3145","phone":"0411111333
","email":"mlong@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT1004
","year":"2013","semester":2,"mark":90,"grade":"HD"},{"unitcode":"F
IT1040","year":"2013","semester":1,"mark":80,"grade":"HD"},{"unitco
de":"FIT2077","year":"2013","semester":2,"mark":72,"grade":"D
"},{"unitcode":"FIT1004","year":"2013","semester":1,"mark":90,"grad
e":"HD"}]},
{"_id":11111113,"name":"Andy Lee","contactInfo":{"address":"1 King
Road, Caulfield South, VIC , 3166","phone":"0411111444
","email":"alees@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT2077
","year":"2013","semester":2,"mark":67,"grade":"C
"},{"unitcode":"FIT1004","year":"2013","semester":1,"mark":72,"grad
e":"D
"},{"unitcode":"FIT1040","year":"2013","semester":1,"mark":74,"grad
e":"D "}]},
{"_id":11111114,"name":"Rani Dewa","contactInfo":{"address":"12/1
Princess Highway, Caulfield East, VIC , 3145","phone":"0411111555
","email":"rdewa@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT1040
","year":"2013","semester":1,"mark":60,"grade":"C
"},{"unitcode":"FIT1004","year":"2013","semester":2,"mark":90,"grad
e":"HD"},{"unitcode":"FIT1004","year":"2013","semester":1,"mark":35
,"grade":"N "}]},
{"_id":11111115,"name":"David
Dumbledore","contactInfo":{"address":"1 Queen Avenue, Caulfield
East, VIC , 3145","phone":"0411111666
","email":"dsmith@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT104
0","year":"2013","semester":1,"mark":80,"grade":"HD"},{"unitcode":"
FIT2077","year":"2013","semester":2,"mark":45,"grade":"N "}]},
{"_id":11111116,"name":"John Chung","contactInfo":{"address":"12/1
Princess Highway, Caulfield East, VIC , 3145","phone":"0411111777
","email":"jchung@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT104
0","year":"2013","semester":2,"mark":80,"grade":"HD"},{"unitcode":"
FIT1004","year":"2013","semester":2,"mark":65,"grade":"C "}]},
{"_id":11111117,"name":"Jake Ryan","contactInfo":{"address":"11
Derby Crescent, Caulfield East, VIC , 3145","phone":"0411111888
","email":"jryan@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT1040
","year":"2013","semester":2,"mark":67,"grade":"C "}]},
{"_id":11111118,"name":"Theo Gupta","contactInfo":{"address":"12
Princess Highway, Caulfield East, VIC , 3145","phone":"0411111999
```

```
","email":"tgupta@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT100
4","year":"2013","semester":2,"mark":55,"grade":"P
"},{"unitcode":"FIT1040","year":"2013","semester":2,"mark":80,"grad
e":"HD"}]},
{"_id":11111119,"name":"Samuel Nguyen","contactInfo":{"address":"56
Queen Avenue, Caulfield East, VIC , 3145","phone":"0411112222
","email":"snguyen@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT10
40","year":"2013","semester":2,"mark":50,"grade":"P
"},{"unitcode":"FIT1004","year":"2013","semester":2,"mark":90,"grad
e":"HD"}]},
{"_id":11111120,"name":"James Dowe","contactInfo":{"address":"100
Princess Highway, Caulfield East, VIC , 3145","phone":"0411112333
","email":"jdowes@monash.edu"},"enrolmentInfo":[{"unitcode":"FIT100
4","year":"2013","semester":2,"mark":65,"grade":"C
"},{"unitcode":"FIT1040","year":"2013","semester":2,"mark":80,"grad
e":"HD"}]},
])
```

3. Show data
    a. Show all data

    ```
    db.studentenrolment.find().pretty()
    ```

    b. Check how many data has been inserted into the collection

    ```
    db.studentenrolment.find().count()
    ```

    c. Show data for student id = 11111111

    ```
    db.studentenrolment.find({"_id":11111111})
    ```

    d. Show id and name of students who were enrolled in FIT2077

    ```
    db.studentenrolment.find({"enrolmentInfo.unitcode":"FIT2077"},
    {"_id":1,"name":1});
    ```

    e. Show details of students who have name John or Jake

    ```
    db.studentenrolment.find({$or:[{"name":/.*John.*/},{"name":/.*
    Jake.*/}]})
    ```

4. Add/remove data from an array

```
db.studentenrolment.find({"_id":11111111})

db.studentenrolment.update({"_id":11111111},{$push:{"enrolmentInfo"
:{"unitcode":"FIT3176","year":"2014","semester":2,"mark":0,"grade":
"-"}}})

db.studentenrolment.find({"_id":11111111})

db.studentenrolment.update({"_id":11111111},{$pull:{"enrolmentInfo"
:{"unitcode":"FIT3176"}}})

db.studentenrolment.find({"_id":11111111})
```

5. Update Value

Update student id 11111111's name from Mary Smith to Mary Karpov. She also changed her phone number to 0432999242

```
db.studentenrolment.find({"_id":11111111})

db.studentenrolment.updateOne({"_id":11111111}, {$set:{"name" :
"Mary Karpov","contactInfo.phone":"0432999242"}})

db.studentenrolment.find({"_id":11111111})
```

6. Delete

   Delete student id 11111111

```
db.studentenrolment.deleteOne({"_id":11111111})

db.studentenrolment.find().count()
```

12.1.2 MongoDB Create Update and Delete

First, download week12_bigdata.txt file from Moodle. Write necessary SQL statements and MongoDB scripts for **12.1.2 and 12.1.3** questions in that file.

1. Using SQL statement, generate a collection of data using this structure/format

```
{
    "_id": 11111111,
    "name": "Mary Smith",
    "contactInfo": {
      "address": "20/1 Princess Highway, Caulfield East, VIC ,
3145",
      "phone": "0411111222     ",
      "email": "msmith@monash.edu"
    },
    "dob": "01-01-1995",
    "enrolmentInfo": [
      {
        "unitcode": "FIT1004",
        "unitname": "Introduction to Data Management",
        "year": "2013",
        "semester": 1,
        "mark": 65,
        "grade": "C "
      },
      {
        "unitcode": "FIT2077",
        "unitname": "Advanced Data Management",
        "year": "2013",
        "semester": 2,
        "mark": 74,
        "grade": "D "
      },
      {
        "unitcode": "FIT1040",
        "unitname": "Programming Fundamental",
        "year": "2013",
```

```
            "semester": 1,
            "mark": 45,
            "grade": "N "
        },
        {
            "unitcode": "FIT1040",
            "unitname": "Programming Fundamental",
            "year": "2013",
            "semester": 2,
            "mark": 74,
            "grade": "D "
        }
    ]
}
```


2. Insert first 10 data generated by the select statement into MongoDB (if it returns error, try to add max 5 documents at one time)
3. Create a new enrolment for studid 11111111, the student is enrolled in FIT2001 (Systems Development) in 2013 semester 2. Since this is a new enrolment, set mark as 0 and grade as "-"
4. Update the enrolment for studid 11111111 in FIT2001, set the mark to 65 and grade to C
5. Delete the enrolment of student id 11111111 in FIT2001

12.1.3 MongoDB Read

Write db.find() commands for following questions:

1. Retrieve data for student id = 11111112
2. Show id of students who have any enrolment mark > 80 (hint: use $gt:80)
3. Retrieve name and contact info of students who enrolled in unit which has "Advanced" as part of its name
4. Retrieve id and name of student who have grade N and P

12.2 SETU and Assignment 2

Spend the rest of the time of this tutorial to fill up the SETU and continue working on your assignment 2.

Important

You need to get into the habit of establishing this as a standard FIT9132 workflow - Pull at the start of your working session, work on the activities you wish to/are able to complete during this session, add (stage), commit changes and then Push the changes back to the FIT GitLab server