National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Computing

CS-213: Advanced Programming

Class: BSCS 7AB

Lab 06: Node.js MongoDB

Date: 3rd October, 2019

Time: 10:00-01:00pm & 02:00-05:00pm

Umaid Zaffar

237560

BSCS 7B

Instructor: Dr. Sidra Sultana

Lab Engineer: Ms. Ayesha Asif

Lab 06: Node.js MongoDB

Introduction

Node.js can be used in database applications. One of the most popular NoSQL database is MongoDB.

Objectives

This lab will get you familiar with the node.js MongoDB environment.

Tools/Software Requirement

Node.js, Notepad

Description

To be able to experiment with the code examples, you will need access to a MongoDB database.

You can download a free MongoDB database at https://www.mongodb.com

Install MongoDB Driver

Let us try to access a MongoDB database with Node.js.

To download and install the official MongoDB driver, open the Command Terminal and execute the following:

Download and install mongodb package:

C:\Users*Your Name*>npm install mongodb

Now you have downloaded and installed a mongodb database driver.

Node.js can use this module to manipulate MongoDB databases:

```
var mongo = require('mongodb');
```

Helping Material

Slides of Lecture 7

https://www.w3schools.com/nodejs/nodejs_mongodb.asp

Lab Tasks

Task 1: Create a database named "mydb". Save the code in a file called "demo_create_mongo_db.js" and run the file.

Task 2: Create a collection called "customers". Save the code in a file called "demo_mongodb_createcollection.js" and run the file.

Task 3: Insert a document in the "customers" collection. Save the code in a file called "demo_mongodb_insert.js" and run the file.

Task 4: Insert multiple documents in the "customers" collection. Save the code in a file called "demo_mongodb_insert_multiple.js" and run the file.

Task 5: Insert three records in a "products" table, with specified _id fields. Save the code in a file called "demo_mongodb_insert_id.js" and run the file.

Task 6: Find the first document in the customer's collection. Save the code in a file called "demo_mongodb_findone.js" and run the file.

Task 7: Return the fields "name" and "address" of all documents in the customers collection

Task 8: Find documents with the address "Park Lane 38". Save the code in a file called "demo_mongodb_query.js" and run the file.

Task 9: Sort the result alphabetically by name. Save the code in a file called "demo_sort.js" and run the file.

Task 10: Delete the document with the address "Mountain 21". Save the code in a file called "demo_delete.js" and run the file.

Task 11: Delete all documents were the address starts with the letter "O". Save the code in a file called "demo_delete_many.js" and run the file

Task 12: Delete the "customers" table. Save the code in a file called "demo_drop.js" and run the file.

Task 13: Update the document with the address "Valley 345" to name="Mickey" and address="Canyon 123". Save the code in a file called "demo_update_one.js" and run the file

Task 14: Consider you have a "customers" collection. Limit the result to only return 5 documents. Save the code above in a file called "demo_mongodb_limit.js" and run the file.

Task 15: Practice the Join operations on different tables.

Solution

Task 1 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, function(err, db){
        if (err) throw err;
        console.log("Database Created");
        db.close();
})
```

Task 1 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Database Created
```

Task 2 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        dbo.createCollection("customers", function(err, res){
```

```
            if (err) throw err;

            console.log("Collection Created");

            db.close();

    })

})
```

Task 2 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Collection Created
```

Task 3 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){

    var dbo = db.db("mydb");

    var obj = {name:"Umaid Zaffar", address:"Islambad"};

    dbo.collection("customers").insertOne(obj, function(err, res){

            if (err) throw err;

            console.log("Document Inserted!");

            db.close();

    })

})
```

Task 3 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Document Inserted!
```

Task 4 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var obj = [
        {name:"Umaid Zaffar", address:"Islambad"},
        {name:"Marium Aslam",address:"Banigala"},
        {name:"Sabayna Ali", address:"Islamabad"},
        {name:"Immad Amir", address:"Lahore"}]
        dbo.collection("customers").insertMany(obj, function(err, res){
                if (err) throw err;
                console.log("Records Inserted:" + res.insertedCount);
                db.close();
        })
})
```

Task 4 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Records Inserted:4
```

Task 5 Code:

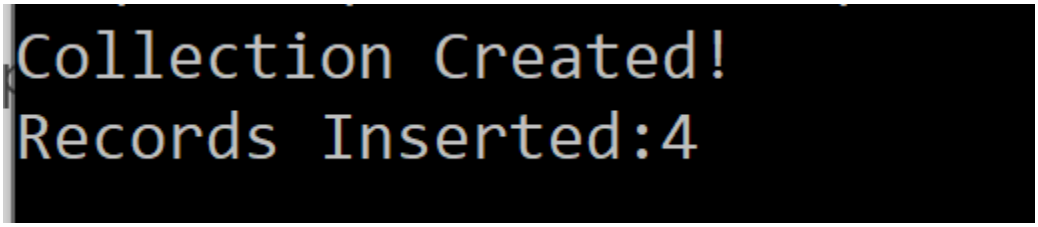
```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var obj = [
        {name:"Umaid Zaffar", _id:0},
        {name:"Marium Aslam",_id:1},
        {name:"Sabayna Ali", _id:2},
        {name:"Immad Amir", _id:3}]
        dbo.createCollection("products", function(err, resu){
                if (err) throw err;
                console.log('Collection Created!')
                dbo.collection("products").insertMany(obj, function(err, res){
                        if (err) throw err;
                        console.log("Records Inserted:" + res.insertedCount);
                        db.close();
                })
```

```
        })
})
```

Task 5 Output Screenshot:



Task 6 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");


        dbo.collection("customers").findOne({}, function(err, res){
                if (err) throw err;
                console.log(res.name);
                db.close();
        })
})
```

Task 6 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Company Inc
```

Task 7 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");


        dbo.collection("customers").find({},{projection:{_id:0, name:1, address:1}}).toArray(function(err,
res){
                if (err) throw err;
                console.log(res);
                db.close();
        })
})
```

Task 7 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
[ { name: 'Company Inc', address: 'Highway 37' },
  { name: 'Umaid Zaffar', address: 'Islambad' },
  { name: 'Umaid Zaffar', address: 'Islambad' },
  { name: 'Marium Aslam', address: 'Banigala' },
  { name: 'Sabayna Ali', address: 'Islamabad' },
  { name: 'Immad Amir', address: 'Lahore' } ]

C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>
```

Task 8 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var query = {address: "Lahore"}
        dbo.collection("customers").find(query).toArray(function(err, res){
                if (err) throw err;
                console.log(res);
                db.close();
        })
})
```

Task 8 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects
[ { _id: 5d963f14b63dd114dc03e8c1,
    name: 'Immad Amir',
    address: 'Lahore' } ]
```

Task 9 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var sort = {name: 1}
        dbo.collection("customers").find().sort(sort).toArray(function(err, res){
                if (err) throw err;
                console.log(res);
                db.close();
        })
})
```

Task 9 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
[ { _id: 5d94d626eae6c43958c49634,
    name: 'Company Inc',
    address: 'Highway 37' },
  { _id: 5d963f14b63dd114dc03e8c1,
    name: 'Immad Amir',
    address: 'Lahore' },
  { _id: 5d963f14b63dd114dc03e8bf,
    name: 'Marium Aslam',
    address: 'Banigala' },
  { _id: 5d963f14b63dd114dc03e8c0,
    name: 'Sabayna Ali',
    address: 'Islamabad' },
  { _id: 5d963e17edb04f184ccbfed7,
    name: 'Umaid Zaffar',
    address: 'Islambad' },
  { _id: 5d963f14b63dd114dc03e8be,
    name: 'Umaid Zaffar',
    address: 'Islambad' } ]
```

Task 10 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var query = {address: "Lahore"}
        dbo.collection("customers").deleteOne(query, function(err, res){
                if (err) throw err;
                console.log(res);
                db.close();
        })
})
```

Task 10 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node
CommandResult {
  result: { n: 1, ok: 1 },
  connection:
   Connection {
     _events:
```

Task 11 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var query = {address: /^I/}
        dbo.collection("customers").deleteMany(query, function(err, res){
                if (err) throw err;
                console.log(res.result.n);
                db.close();
        })
})
```

Task 11 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
3
```

Task 12 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var query = {address: /^I/}
        dbo.collection("customers").drop( function(err, res){
                if (err) throw err;
                if(res) console.log("Deleted");
                db.close();
        })
})
```

Task 12 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Deleted
```

Task 13 Code:

```
var MongoClient = require("mongodb").MongoClient;
```

```
var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
        var dbo = db.db("mydb");
        var oldValue = {name: "Marium Aslam"}
        var newValue = {$set:{_id:10, name:"Marium"}}
        dbo.collection("customers").updateOne(oldValue, newValue, function(err, res){
                if (err) throw err;
                console.log("Updated");
                db.close();
        })
})
```

Task 13 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
Updated
```

Task 14 Code:

```
var MongoClient = require("mongodb").MongoClient;


var url = "mongodb://localhost:27017/mydb";


MongoClient.connect(url, { useNewUrlParser: true ,useUnifiedTopology: true}, function(err, db){
```

```
        var dbo = db.db("mydb");


        dbo.collection("customers").find().limit(5).toArray(function(err, res){
            if (err) throw err003B
            console.log(res);
            db.close();
        })
    })
```

Task 14 Output Screenshot:

```
C:\Users\Umaid Zaffar\Documents\Node_Projects\mongodb>node first.js
[ { _id: 5d99786e7ce09d6574fdd091,
    name: 'John',
    address: 'Highway 71' },
  { _id: 5d99786e7ce09d6574fdd092,
    name: 'Peter',
    address: 'Lowstreet 4' },
  { _id: 5d99786e7ce09d6574fdd093,
    name: 'Amy',
    address: 'Apple st 652' },
  { _id: 5d99786e7ce09d6574fdd094,
    name: 'Chuck',
    address: 'Main Road 989' },
  { _id: 5d99786e7ce09d6574fdd095,
    name: 'Viola',
    address: 'Sideway 1633' } ]
```

Task 15 Code:

```
MongoClient.connect(url, function(err, db) {
  if (err) throw err;
  var dbo = db.db("mydb");
  dbo.collection('orders').aggregate([
```

```
{ $lookup:
  {
    from: 'products',
    localField: 'product_id',
    foreignField: '_id',
    as: 'orderdetails'
  }
 }
]).toArray(function(err, res) {
if (err) throw err;
console.log(JSON.stringify(res));
db.close();
});
});
```

Task 15 Output Screenshot:

Deliverables

Compile a single word document by filling in the solution part and submit this Word file on LMS. This lab grading policy is as follows: The lab is graded between 0 to 10 marks. The submitted solution can get a maximum of 5 marks. At the end of each lab or in the next lab, there will be a viva related to the tasks. The viva has a weightage of 5 marks. Insert the solution/answer in this document. You must show the implementation of the tasks in the designing tool, along with your complete Word document to get your work graded. You must also submit this Word document on the LMS. In case of any problems with submissions on LMS, submit your Lab assignments by emailing it to Ms. Ayesha Asif: ayesha.asif@seecs.edu.pk.