**Exercise 9 – Dynamic Website using NodeJS and Express**

**Name:** Badri MSV **Roll No:** 195002017

**Subject:** UIT1611 – Web Programming Lab **Faculty:** Dr. S. Sasirekha

**Date:** 17/05/2022

**Aim:**

To create a dynamic website using Node and Express.

**Procedure:**

* Connect to mongoDB using node.js
  + var MongoClient = require('mongodb').MongoClient;
  + var url = "mongodb://localhost:27017/";
  + MongoClient.connect(url,function(err,db{});
* On unsuccessful connection, throw error.
  + if (err) throw err;
* Get input from user using prompt to for fields to insert.
  + const prompt = require('prompt');
  + prompt.start();
* Using db.insertOne(), insert the record and on unsuccessful insertion, throw error.
* For deletion, get the value of the record based on which record you want to delete.
* Switch to the particular database.
* Define a variable which has the value to be deleted.
* Using db.deleteOne(), delete the record by passing the variable as a query.
* On unsuccessful deletion, throw error
* If the options do not match, print Invalid.
* For updation, get the value of the record based on which record you want to update.
* Using db.updateOne(),update the given record, if not available print invalid.
* To display the database use db.find().

**index.js:**

var MongoClient = require('mongodb').MongoClient;

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

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

    if (err) throw err;

    var dbo = db.db("dynamic");

    const prompt = require('prompt-sync')({ sigint: true });

    console.log("MENU\n1.Insert\n2.Update\n3.Delete\n4.Display");

    var option = prompt('Enter the option : ');

    switch (option) {

        case '1':

            var no\_of\_students = prompt('Enter number of students: ');

            var myobj = [];

            var iter = 0;

            for (iter = 0; iter < no\_of\_students; iter++) {

                var name = prompt("Enter the Name : ");

                var clg\_name = prompt("Enter your college name: ");

                myobj.push({ name: name, clg\_name: clg\_name });

            }

            dbo.collection("demo").insertMany(myobj, function (err, res) {

                console.log("Inserting data :");

                if (err) throw err;

                console.log("Number of document(s) inserted : " + res.insertedCount + "\n");

                db.close();

            });

            break;

        case '2':

            var u\_name = prompt("Enter the name to be updated : ");

            var myquery = { name: u\_name };

            var u\_clg\_name = prompt("Enter the new college name: ");

            var newvalues = { $set: { clg\_name: u\_clg\_name } };

            dbo.collection("demo").updateMany(myquery, newvalues, function (err, res) {

                console.log("\nUpdating data :");

                if (err) throw err;

                if (res.result.nModified == 0) {

                    console.log('No match found!\n');

                }

                else {

                    console.log("Number of document(s) updated : " + res.result.nModified + "\n");

                }

                db.close();

            });

            break;

            break;

        case '3':

            console.log('\tDELETE MENU:\n\t1.Delete specific data\n\t2.Delete all data')

            var del\_option = prompt('\tSelect an option : ');

            switch (del\_option) {

                case '1':

                    var del\_clg\_name = prompt('Enter the college name to be deleted : ');

                    var myquery = { clg\_name: del\_clg\_name };

                    dbo.collection("demo").deleteMany(myquery, function (err, obj) {

                        if (err) throw err;

                        if (obj.result.n == 0) {

                            console.log('No match found!\n');

                        }

                        else {

                            console.log("Number of document(s) deleted : " + obj.result.n + "\n");

                        }

                        db.close();

                    });

                    break;

                case '2':

                    dbo.collection("demo").deleteMany(function (err, res) {

                        console.log("\nDeleting data :");

                        if (err) throw err;

                        if (res.result.n == 0) {

                            console.log('Collection is empty!\n')

                        }

                        else {

                            console.log("Number of document(s) deleted : " + res.result.n + "\n");

                        }

                        db.close();

                    });

                    break;

                default:

                    console.log('Enter valid option.\n');

                    db.close();

            }

            break;

        case '4':

            dbo.collection("demo").find({}, { projection: { \_id: 0, name: 1, clg\_name: 1 } }).toArray(function (err, result) {

                console.log("\nDisplaying data :");

                if (err) throw err;

                // console.log(result);

                if (result.length == 0) {

                    console.log('Collection is empty!\n');

                }

                else {

                    var i;

                    for (i = 0; i < result.length; i++) {

                        console.log("Name : " + result[i].name + ", College name : " + result[i].clg\_name);

                    }

                    console.log('\n');

                }

                db.close();

            });

            break;

        default:

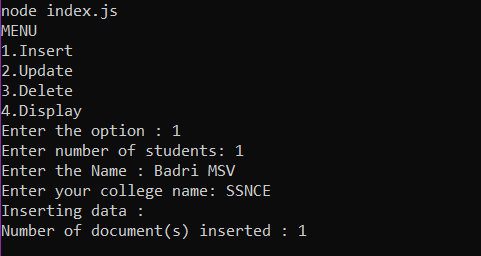
            console.log('Enter valid input.\n');

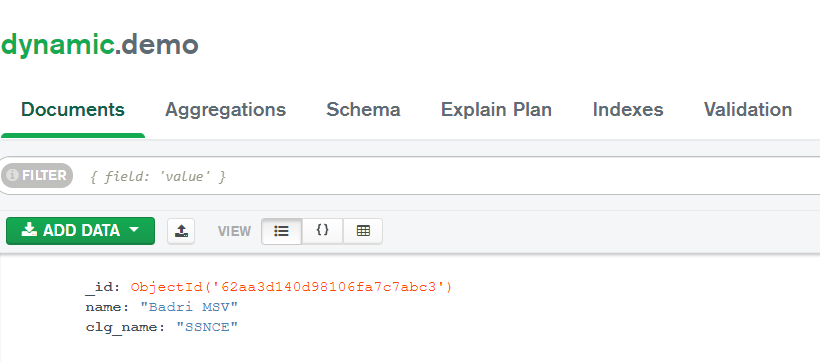
            db.close();

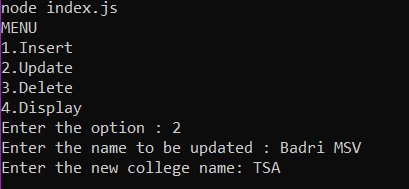
    }

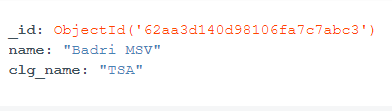
});

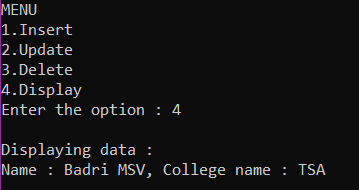
**Output:**

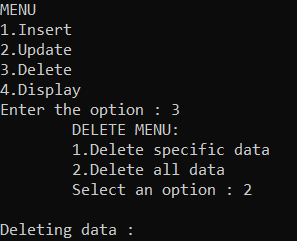


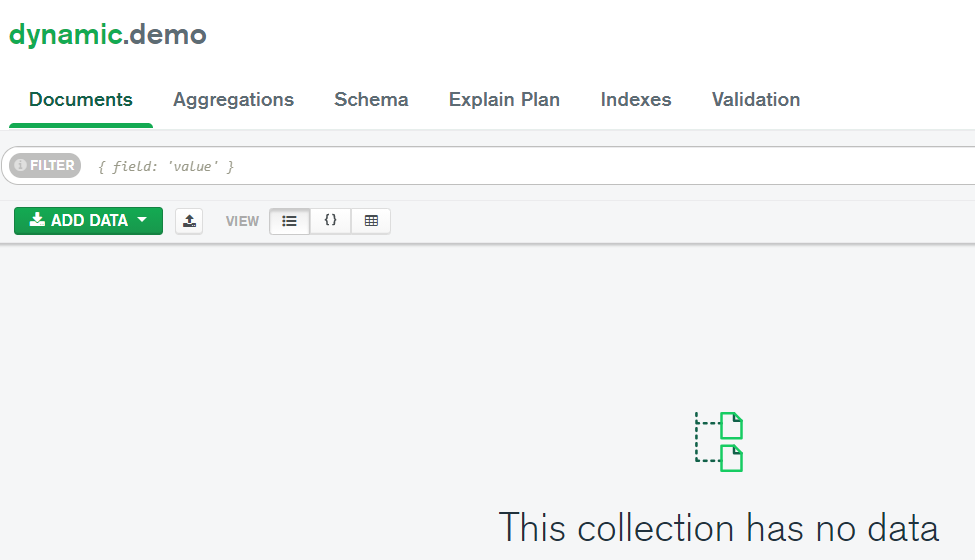


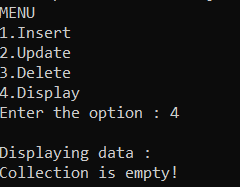












**Result:**

A dynamic website using NodeJS and Express was created successfully.