Task – 28/09/2020

**Task 1:**

**// Using NodeJs Insert 5 records inside the collection use insert many**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  var objarray = [

      {name: "Tarak", address: "Mumbai"},

      {name: "Yukti", address: "Hyderabad"},

      {name: "Mihir", address: "Bangalore"},

      {name: "Rishi", address: "Bangalore"},

      {name: "Sriya", address: "Mumbai"}

  ]

  dbo.collection("customers").insertMany(objarray, function(err, res) {

    if (err) throw err;

    console.log("docs inserted");

    db.close();

  });

});

**Task 2:**

**// Show all records**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find().toArray(function(err, result) {

    if (err) throw err;

    console.log(result);

    db.close();

  });

});

**Task 3:**

**// Show all records with given name**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find({name:"Anurag"}).toArray(function(err, result) {

    if (err) throw err;

    console.log(result);

    db.close();

  });

});

**Task 4:**

**//Using NodeJS show Name & Address where Name starts with "S"**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find({name:{ $regex: /^S/ }}).toArray(function(err, result) {

    if (err) throw err;

    console.log(result);

    db.close();

  });

});

**Task 5:**

**// Update docs by adding prefix to names with given condition**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  var q = {address: /^H/ }

  var sq = [{ $set : { name : {$concat : ["Dr. ","$name"]}}}]

  dbo.collection("customers").updateMany(q,sq,function(err, res) {

      if (err) throw err;

      console.log("updated!!");

      console.log(res)

      db.close();

    }

  );

});

**Task 6:  
 (a)**

**//Using NodeJs Show All Database existing inside the MongoDB**

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

var dbName = 'test';

mongoClient.connect("mongodb://localhost/",

{useUnifiedTopology: true,useNewUrlParser:true, poolSize: 5, reconnectInterval: 500 },

function(err,client){

    if(err) throw err

    var dbAdmin=client.db(dbName).admin();

    dbAdmin.listDatabases(function (err,databases) {

        if (err) throw err

        console.log("before adding databases");

        console.log(databases);

        client.close();

    })

});

**(b)**

**//Show All Collection existing inside the mydb database**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.listCollections().toArray(function(err,res){

      if(err) throw err

      console.log(res)

      db.close()

  });

});

**(c)**

**//Show Number Of Documents existing inside the Customers collections**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find().count(function(err,res){

      if(err) throw err

      console.log(res)

      db.close()

  })

});

**(d)**

**//Show All records of Customers & Sort the result alphabetically by name.**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find().sort({name:1}).toArray(function(err,res){

      if(err) throw err

      console.log(res)

      db.close()

  })

});

**Task 7:**

**//Limit the result to show only 5 documents**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  dbo.collection("customers").find().limit(5).toArray(function(err, result) {

    if (err) throw err;

    console.log(result);

    db.close();

  });

});

**Task 8:**

**//Task 8 : Using NodeJS delete the record where name starts with "A"**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  var q = {name: /^A/}

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

      if (err) throw err

      console.log(res)

      db.close()

  })

});

**Task 9:**

**//Using Node js add one new field as "Salary" & Insert the Salary for all customers**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  var sq = [{ $set : { salary:3000 }}]

  dbo.collection("customers").updateMany({},sq,function(err, res) {

      if (err) throw err;

      console.log("updated!!");

      console.log(res)

      db.close();

    }

  );

});

**Task 10:**

**//Using Nodejs Show all customers getting the Salary between 20000 and 40000**

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

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

MongoClient.connect(url, function(err, db) {

  if (err) throw err;

  var dbo = db.db("mydb2");

  var q = {$and:[{salary:{$gte:20000}},{salary:{$lte:40000}}]}

  dbo.collection("customers").find(q).toArray(function(err, result) {

    if (err) throw err;

    console.log(result);

    db.close();

  });

});

**Task 11:**

**// Using NodeJS create a new collection tempcollection & drop the Collection tempcollection**

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

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

// MongoClient.connect(url, function(err, db) {

//   if (err) throw err;

//   var dbo = db.db("mydb2");

//   dbo.createCollection("tempcollection", function(err, res) {

//     if (err) throw err;

//     console.log("Collection created!");

//     db.close();

//   });

// });

MongoClient.connect(url, function(err, db) {

    if (err) throw err;

    var dbo = db.db("mydb2");

    dbo.dropCollection("tempcollection", function(err, res) {

      if (err) throw err;

      console.log("Collection deleted");

      db.close();

    });

  });