**EXPERIMENT 4**

**IMPLEMENTATION:**

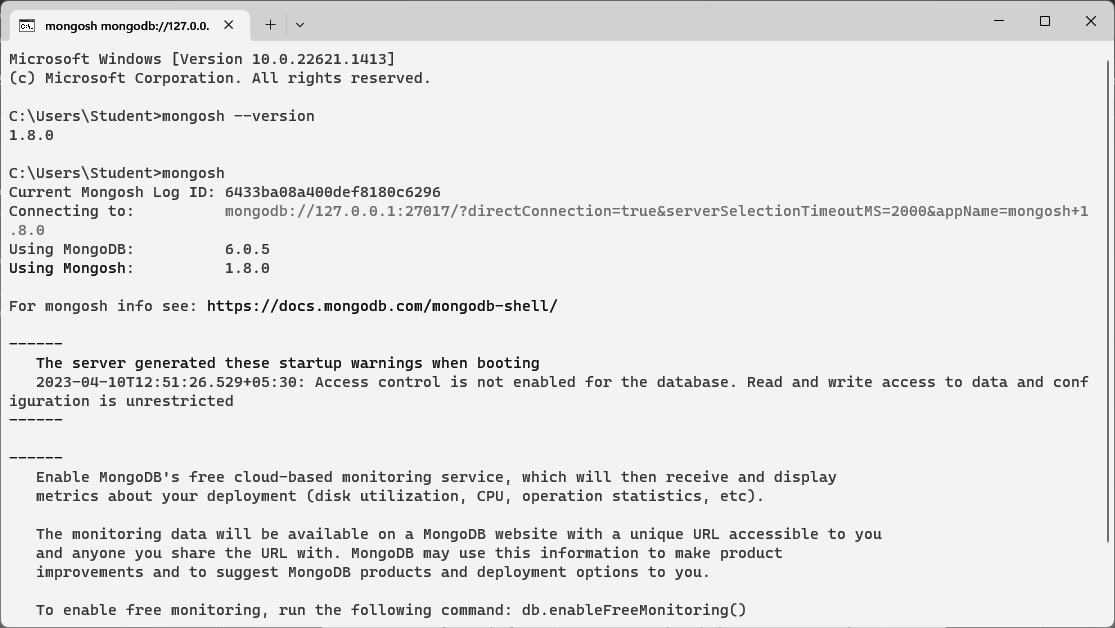
**MONGODB**

**Installation:**

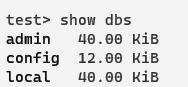
**Link: https://www.mongodb.com/try/download/community**

**Link: https://www.mongodb.com/try/download/shell**

Checking the version



Show all the databases



Create a new database if already does not exist



Create a new collection

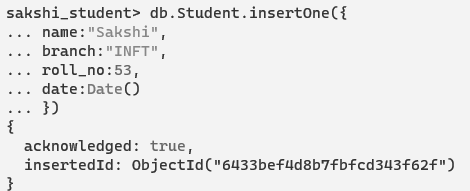


Show all the collections existing



INSERT OPERATION

1. Insert one



1. Insert many



SEARCH OPERATION







DELETE

1. Delete one

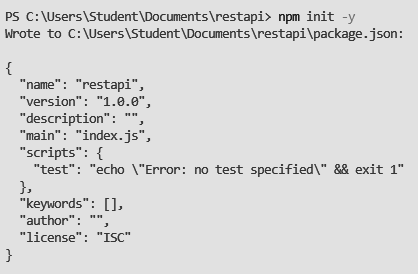


1. Delete many



**RESTFUL API USING MONGODB**

1. Initialize NodeJs project with npm init



1. Installation:
2. npm i express
3. npm i mongoose@6.10.2
4. npm install -g nodemon
5. npm i body-parser
6. Install thunder client/Postman
7. Create files: index.js, contactModel.js, contactcontroller.js, api-routes.js

**index.js**

// Import express

let express = require('express');

// Import Body parser

let bodyParser = require('body-parser');

// Import Mongoose

let mongoose = require('mongoose');

// Initialise the app

let app = express();

// Import routes

let apiRoutes = require("./api-routes");

// Configure bodyparser to handle post requests

app.use(bodyParser.urlencoded({

extended: true

}));

app.use(bodyParser.json());

// Connect to Mongoose and set connection variable

mongoose.connect('mongodb://localhost:27017/phone', { useNewUrlParser: true });

var db = mongoose.connection;

// Added check for DB connection

if (!db)

console.log("Error connecting db")

else

console.log("Db connected successfully")

// Setup server port

var port = process.env.PORT || 8080;

// Send message for default URL

app.get('/', (req, res) => res.send('Hello World with Express'));

// Use Api routes in the App

app.use('/api', apiRoutes);

// Launch app to listen to specified port

app.listen(port, function() {

console.log("Running RestHub on port " + port);

});

**contactModel.js**

var mongoose = require('mongoose'); // Setup schema

var contactSchema = mongoose.Schema({

name: {

type: String,

required: true

},

email: {

type: String,

required: true

},

gender: String,

phone: String,

create\_date: {

type: Date,

default: Date.now

}

}); // Export Contact model

var Contact = module.exports = mongoose.model('contact', contactSchema);

module.exports.get = function(callback, limit) {

Contact.find(callback).limit(limit);

}

**Contactcontroller.js**

// contactController.js// Import contact model

Contact = require('./contactModel'); // Handle index actions

exports.index = function(req, res) {

Contact.get(function(err, contacts) {

if (err) {

res.json({

status: "error",

message: err,

});

}

res.json({

status: "success",

message: "Contacts retrieved successfully",

data: contacts

});

});

}; // Handle create contact actions

exports.new = function(req, res) {

var contact = new Contact();

contact.name = req.body.name ? req.body.name : contact.name;

contact.gender = req.body.gender;

contact.email = req.body.email;

contact.phone = req.body.phone; // save the contact and check for errors

contact.save(function(err) {

if (err)

res.json(err);

res.json({

message: 'New contact created!',

data: contact

});

});

}; // Handle view contact info

exports.view = function(req, res) {

Contact.findById(req.params.contact\_id, function(err, contact) {

if (err)

res.send(err);

res.json({

message: 'Contact details loading..',

data: contact

});

});

}; // Handle update contact info

exports.update = function(req, res) {

Contact.findById(req.params.contact\_id, function(err, contact) {

if (err)

res.send(err);

contact.name = req.body.name ? req.body.name : contact.name;

contact.gender = req.body.gender;

contact.email = req.body.email;

contact.phone = req.body.phone; // save the contact and check for errors

contact.save(function(err) {

if (err)

res.json(err);

res.json({

message: 'Contact Info updated',

data: contact

});

});

});

}; // Handle delete contact

exports.delete = function(req, res) {

Contact.remove({

\_id: req.params.contact\_id

}, function(err, contact) {

if (err)

res.send(err);

res.json({

status: "success",

message: 'Contact deleted'

});

});

};

**api-routes.js**

// api-routes.js// Initialize express router

let router = require('express').Router(); // Set default API response

router.get('/', function(req, res) {

res.json({

status: 'API Its Working',

message: 'Welcome to RESTHub crafted with love!',

});

}); // Import contact controller

var contactController = require('./contactController'); // Contact routes

router.route('/contacts')

.get(contactController.index)

.post(contactController.new);

router.route('/contacts/:contact\_id')

.get(contactController.view)

.patch(contactController.update)

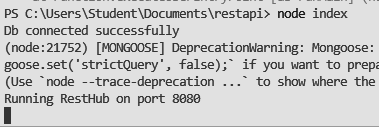
.put(contactController.update)

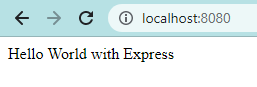
.delete(contactController.delete); // Export API routes

module.exports = router;

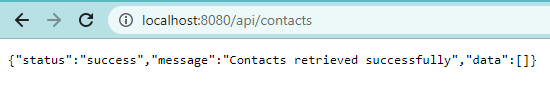
1. Check port 8080 to see if it's running.

Command: node index

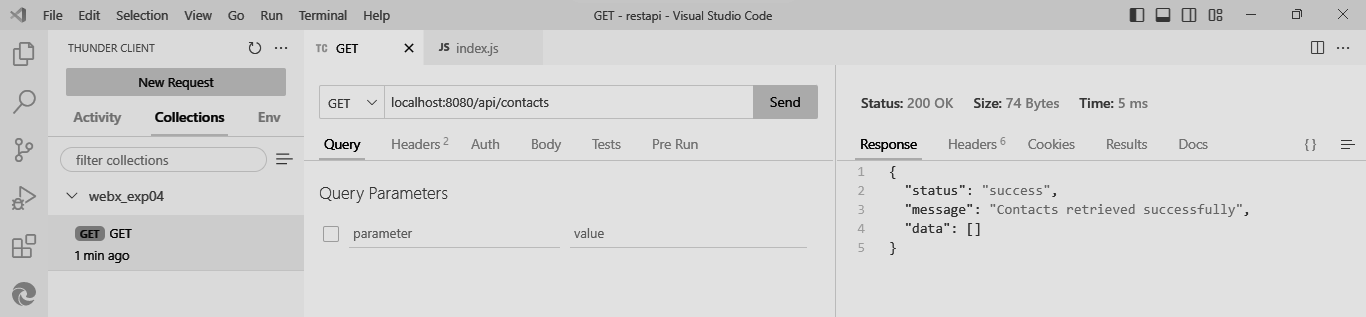




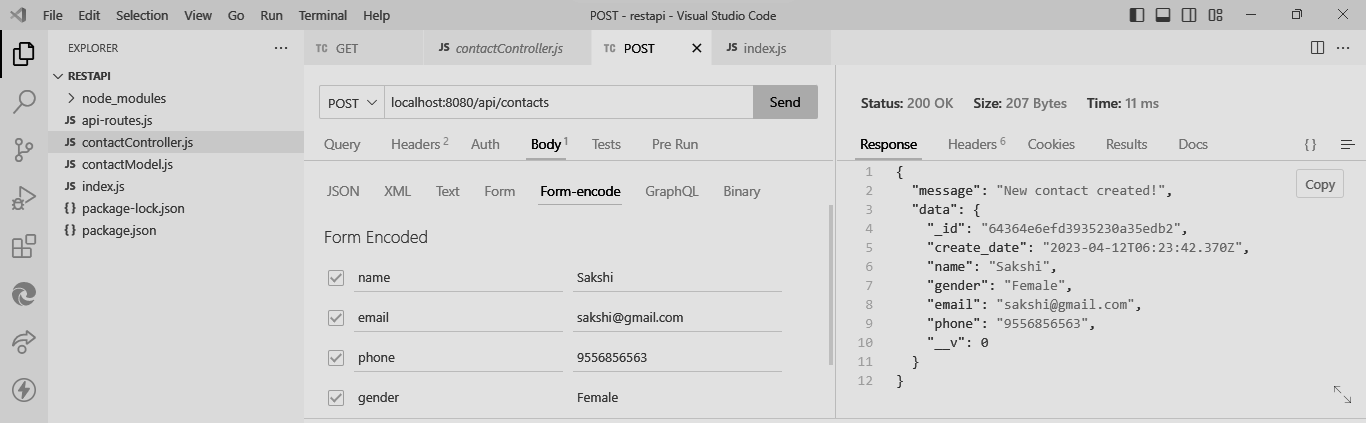
1. Go to localhost:8080/api/contacts

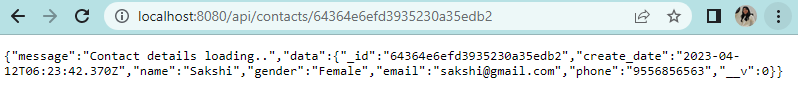


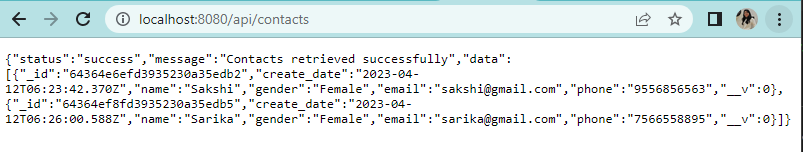
1. GET contacts



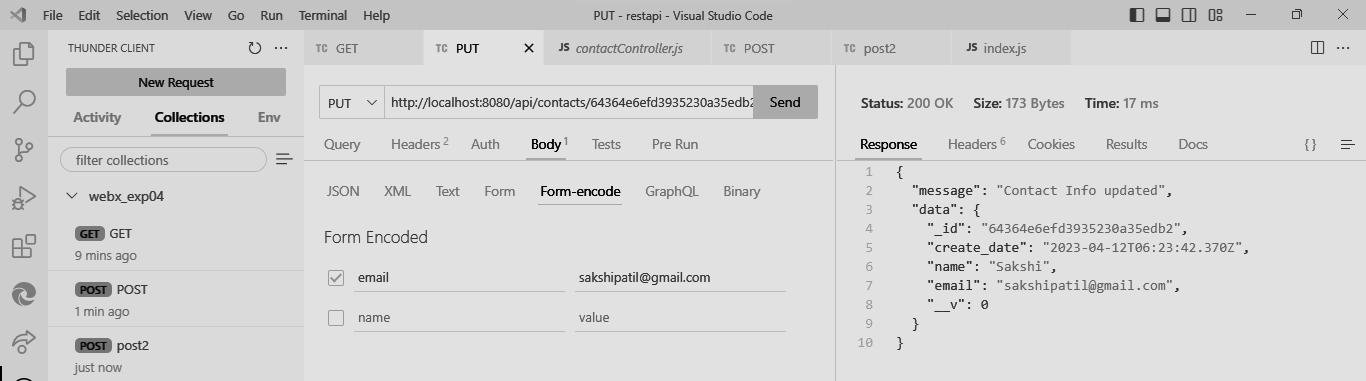
1. POST Contacts



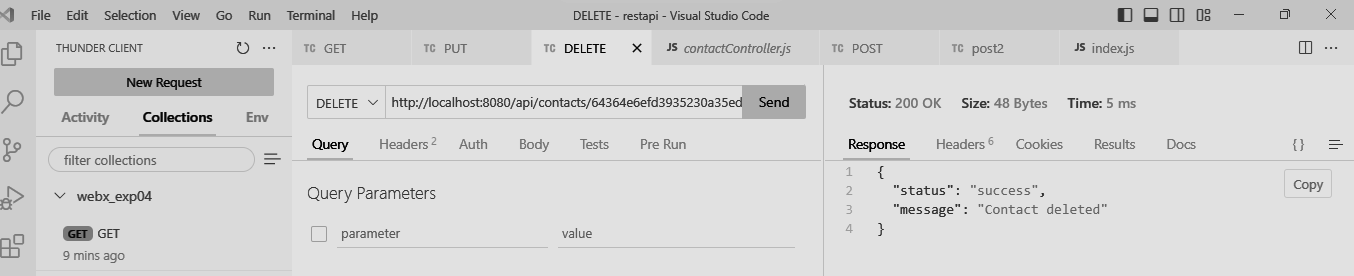




1. PUT Contact



1. DELETE Contact



1. Finally, after all the steps above, the database will look like this.

