Database Queries:

For testing the server and database connection, POSTMAN was used. The test data resembles the following: (contactId is the primary key, if “0” is entered in this field, the primary key will autoincrement.) To verify the correct operation of the software POST operation, it is best to open the SQL-SERVER and right click on the table then select: ‘Select the top 1000 rows”.

{

                "contactId": 0,

                "firstName": "333",

                "lastName": "333",

                "emailAddr": "beta@my.com",

                "street": "40 West Main",

                "city": "Paducah",

                "stateProvince": "Kentucky",

                "countryCode": "USA",

                "postalCode": 88243,

                "homephone": "1112223333",

                "workphone": "2223334444",

                "cellphone": "3334445555",

                "messageHolder": "This is the first entry"

            }

A word on using the console to print the Http message as it is received.

With the terminal, install the *morgan* package. “npm install morgan - - save”

In server.js import morgan by entering const morganMonitor = require('morgan');

And deploy it by entering app.use(morganMonitor('tiny'));

As each Http message is received, its method will be printed in the terminal window.

Below is an example of what will be seen when a ‘post’ message is processed by the server.

POST /contacts 201 2 - 57.356 ms

The updated working ‘server.js’ is shown below:

const express = require('express');   // import express

const cors = require('cors');      // import   cors

const allroutes = require('./routes/allroutes');

const morganMonitor = require('morgan');

const API\_PORT = process.env.PORT || 5000;  // Run the server on Port 5000

const app = express();

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

app.use(cors());  // activate   CORS

app.use(morganMonitor('tiny'));

app.use( bodyParser.urlencoded({

extended: true,})

)

app.use(bodyParser.json())

// activate router for the api's

app.use('/',allroutes);

app.listen(API\_PORT, (req, res)=> {

    console.log(`LISTENING on port ${API\_PORT}`);

});