Node.js Express Server Application

Express is a minimal and flexible Node.js web application framework that provides a robust set of features to develop web and mobile applications. It facilitates the rapid development of Node based Web applications.

Software Requirement:

- VS Code
- Node.js – It is a server-side platform built on Google Chrome's JavaScript Engine (V8 Engine).

Procedure:

1. Create Node Project Folder.
2. Go to Project Folder and type **code .** command to open project in VS Code
3. Generate **package.json**:
   npm init -y (to create node workspace/ to setup node project)
4. Installing Express.js and its Dependencies:
   npm install express – Web Framework
5. Create **index.js** file and Write "Hello World" JavaScript Code
6. Run Project: node index.js

Index.js

```
// npm install express
var express = require('express');
var app = express();

var bodyParser = require("body-parser");
app.use(bodyParser.urlencoded({extended:true}));
app.use(bodyParser.json());

// Default Route
app.get('/', function (req, res) {
   res.send('<h1>Hello World</h1>');
})

// set port, listen for requests
const PORT = process.env.PORT || 8080;

app.listen(PORT, () => {
   console.log(`Server is running on port ${PORT}.`);
});
```

```
//Connect to MS SQL Server ----------------------------
//npm install mssql - SQL Server Database client for node

var sql = require("mssql");
var dbConfig = {
   user: "Intern",
   password: "datalabs@123",
   server: "38.17.52.214",
   database: "OpenData",
   port: 1433,
   synchronize: true,
   trustServerCertificate: true,
};

// URL: http://localhost:8080/mssql/GetAMCList
app.get("/mssql/GetAMCList", function (req, res) {
   var dbConn = new sql.ConnectionPool(dbConfig);

   dbConn.connect().then(function () {
      var request = new sql.Request(dbConn);

      request.query( "select * from AMC" ).then(function (resp) {
         dbConn.close();
         res.status(200).json({
            status: "success",
            length: resp.recordset?.length,
            data: resp.recordset,
         });
      }).catch(function (err) {
         console.log(err);
         dbConn.close();
      });
   }).catch(function (err) {
      console.log("error " + err);
   });
})
```

```
// URL: http://localhost:8080/mssql/GetYardList
app.get("/mssql/GetYardList", function (req, res) {
   var dbConn = new sql.ConnectionPool(dbConfig);

   dbConn.connect().then(function () {
      var request = new sql.Request(dbConn);
      request.execute("GetYardList").then(function (resp) {
         dbConn.close();
         res.status(200).json({
            status: "success",
            length: resp.recordset?.length,
            data: resp.recordset,
         });
      }).catch(function (err) {
         console.log(err);
         dbConn.close();
      });
   }).catch(function (err) {
      console.log("error " + err);
   });
})

// URL: http://localhost:8080/mssql/GetVaritiesByCommodity?CommCode=1
app.get("/mssql/GetVaritiesByCommodity", function (req, res) {

   var CommCode = req.query.CommCode;
   var dbConn = new sql.ConnectionPool(dbConfig);
   dbConn.connect().then(function () {
      var request = new sql.Request(dbConn);
      request.input('CommCode', sql.Int, CommCode);
      request.execute("GetVaritiesByCommodity").then(function (resp) {
         dbConn.close();
         res.status(200).json({
            status: "success",
            length: resp.recordset?.length,
            data: resp.recordset,
         });
      }).catch(function (err) {
         console.log(err);
         dbConn.close();
      });
   }).catch(function (err) {
      console.log("error " + err);
   });
})

//Run App: node index.js
```