23. Design and develop a responsive website to prepare one semester result of VIT students using JavaScript, React and Node JS and MySQL. Take any four subjects with MSE Marks (30%) ESE Marks (70%).

Step 1: Install Prerequisites

Ensure you have the following installed on your system:

1. **Node.js - Download and install Node.js**
2. **MySQL - Download and install MySQL**

Step 2: Set Up MySQL Database

Open MySQL CLI:

```
Open MySQL CLI:

-- Create database
CREATE DATABASE vit_data;

-- Use the database
USE vit_data;

-- Create table
CREATE TABLE vit_results (
    id INT AUTO_INCREMENT PRIMARY KEY,
    student_name VARCHAR(100) NOT NULL,
    reg_number VARCHAR(15) NOT NULL UNIQUE,
    subject1_mse FLOAT NOT NULL,
    subject1_ese FLOAT NOT NULL,
    subject2_mse FLOAT NOT NULL,
    subject2_ese FLOAT NOT NULL,
    subject3_mse FLOAT NOT NULL,
    subject3_ese FLOAT NOT NULL,
    subject4_mse FLOAT NOT NULL,
    subject4_ese FLOAT NOT NULL
);

-- Insert dummy data
INSERT INTO vit_results (student_name, reg_number, subject1_mse, subject1_ese,
subject2_mse, subject2_ese, subject3_mse, subject3_ese, subject4_mse,
subject4_ese)
VALUES
('Alice Johnson', '21BCE1001', 25, 65, 28, 68, 30, 70, 24, 66),
('Bob Smith', '21BCE1002', 22, 63, 20, 58, 27, 67, 26, 68),
('Charlie Brown', '21BCE1003', 29, 69, 30, 70, 28, 68, 30, 70),
('Diana Prince', '21BCE1004', 20, 60, 25, 65, 21, 61, 22, 62);
```

Step 3: Set Up the Backend (Node.js + Express)

1. **Create Project Folder**:
   - In your desired location, create a new folder for the backend project.
   - Navigate to that folder in the terminal.

```
mkdir backend
cd backend
```

2. **Initialize Node.js Project:**

```
npm init -y
```

3. **Install Required Packages:**

   Install the necessary dependencies:

```
npm install express mysql2 cors body-parser
```

4. **Create server.js:**

Create a server.js file and add the following code to connect to MySQL and expose the results via an API:

```
Create a server.js file and add the following code to connect to MySQL
and expose the results via an API:

const express = require("express");
const mysql = require("mysql2");
const bodyParser = require("body-parser");
const cors = require("cors");

const app = express();
app.use(cors());
app.use(bodyParser.json());

// MySQL Connection
const db = mysql.createConnection({
  host: "localhost",
  user: "root",
  password: "root", // Replace with your actual MySQL password
  database: "vit_data",
});

db.connect((err) => {
  if (err) {
    console.error("Error connecting to database:", err);
    return;
  }
  console.log("Connected to database");
});

// Fetch all student results
app.get("/results", (req, res) => {
  const query = "SELECT * FROM vit_results";
  db.query(query, (err, results) => {
```

```
    if (err) {
      console.error(err);
      res.status(500).send(err);
    } else {
      res.send(results);
    }
  });
});

// Start the server
const PORT = 5001;
app.listen(PORT, () => {
  console.log(`Server running on http://localhost:${PORT}`);
});
```

Step 4: Set Up the Frontend (React)

1. **Create React Project**:

Open a new terminal window and create a React project:

```
npx create-react-app vit-results-frontend
cd vit-results-frontend
npm install axios
```

2. **Create App.js:**

Replace the code in src/App.js with the following code to display the results:

```
Replace the code in src/App.js with the following code to display the
results:

import React, { useEffect, useState } from "react";
import axios from "axios";

const App = () => {
  const [results, setResults] = useState([]);
  const [error, setError] = useState(""); // State to track errors

  useEffect(() => {
    // Fetch results from backend
    axios.get("http://localhost:5001/results") // Corrected port to match
backend
      .then(response => setResults(response.data))
      .catch(error => {
        setError("Error fetching data: " + error.message); // Set error
message
        console.error("Error fetching data:", error);
      });
  }, []);

  return (
    <div style={{ padding: "20px", fontFamily: "Arial, sans-serif" }}>
      <h1>VIT Semester Results</h1>
      {error && <div style={{ color: "red" }}>{error}</div>} {/* Display
error message */}
      <table border="1" cellPadding="10" style={{ borderCollapse:
```

```
"collapse", width: "100%" }}>
        <thead>
          <tr>
            <th>Name</th>
            <th>Reg Number</th>
            <th>Subject 1</th>
            <th>Subject 2</th>
            <th>Subject 3</th>
            <th>Subject 4</th>
          </tr>
        </thead>
        <tbody>
          {results.length === 0 ? (
            <tr>
              <td colSpan="6">No results found</td> {/* Handle empty
results */}
            </tr>
          ) : (
            results.map(result => (
              <tr key={result.reg_number}>
                <td>{result.student_name}</td>
                <td>{result.reg_number}</td>
                <td>{result.subject1_mse * 0.3 + result.subject1_ese *
0.7}</td>
                <td>{result.subject2_mse * 0.3 + result.subject2_ese *
0.7}</td>
                <td>{result.subject3_mse * 0.3 + result.subject3_ese *
0.7}</td>
                <td>{result.subject4_mse * 0.3 + result.subject4_ese *
0.7}</td>
              </tr>
            ))
          )}
        </tbody>
      </table>
    </div>
  );
};

export default App;
```

5. Run the Frontend:

   In the terminal, run the following command to start the React frontend:

```
npm start
```