**Objective:-** Develop a web application which involves database operations using Express

**Code :-**

**Frontend :-**

**App.jsx**

import { useState } from 'react'

import {BrowserRouter, Routes, Route} from 'react-router-dom'

import './Home.jsx'

import './App.css'

import Home from './Home.jsx'

import Create from '../src/Create.jsx'

import Read from './Read.jsx'

import Update from './Update.jsx'

function App() {

  return (

    <>

    <BrowserRouter>

      <Routes>

        <Route path='/' element={<Home/>}></Route>

        <Route path='/create' element={<Create/>}></Route>

        <Route path='/read/:id' element={<Read/>}></Route>

        <Route path='/edit/:id' element={<Update/>}></Route>

      </Routes>

    </BrowserRouter>

    </>

  )

}

export default App

**Create.jsx**

import axios from 'axios';

import React, { useState } from 'react';

import { useNavigate } from 'react-router-dom';

function Create() {

    const [values, setValues] = useState({

        name: '',

        age: '',

        grade: ''

    });

    const navigate=useNavigate();

    const handleSubmit = (e) => {

        e.preventDefault();

        axios.post('http://localhost:8081/student', values)

        .then(res => {

          console.log(res);

          navigate('/')

        })

        .catch(err => console.log(err));

    };

    return (

        <div className="d-flex vh-100 justify-content-center align-items-center bg-light">

            <div className="w-50 bg-white p-4 rounded shadow">

                <h2 className="text-center">Add Student</h2>

                <form onSubmit={handleSubmit}>

                    <div className="mb-3">

                        <label className="form-label">Name</label>

                        <input

                            type="text"

                            placeholder="Enter Name"

                            className="form-control"

                            value={values.name}

                            onChange={e => setValues({...values, name: e.target.value})}

                        />

                    </div>

                    <div className="mb-3">

                        <label className="form-label">Age</label>

                        <input

                            type="text"

                            placeholder="Enter Age"

                            className="form-control"

                            value={values.age}

                            onChange={e => setValues({...values, age: e.target.value})}

                        />

                    </div>

                    <div className="mb-3">

                        <label className="form-label">Grade</label>

                        <input

                            type="text"

                            placeholder="Enter Grade"

                            className="form-control"

                            value={values.grade}

                            onChange={e => setValues({...values, grade: e.target.value})}

                        />

                    </div>

                    <button className="btn btn-primary w-100">Submit</button>

                </form>

            </div>

        </div>

    );

}

export default Create;

**Home.jsx**

import React, { useEffect, useState } from 'react';

import axios from 'axios';

import { Link } from 'react-router-dom';

function Home() {

    const [data, setData] = useState([]);

    const fetchStudents = () => {

      axios.get('http://localhost:8081/')

          .then(res => setData(res.data))

          .catch(err => console.log(err));

    }

    useEffect(() => {

      fetchStudents();

    }, []);

    const handleDelete = (id) => {

      if (window.confirm("Are you sure you want to delete this student?")) {

          axios.post(`http://localhost:8081/delete/${id}`)

              .then(res => {

                  console.log("Deleted Successfully", res.data);

                  fetchStudents(); // Refresh the list

              })

              .catch(err => console.log(err));

      }

  };

    return (

        <div>

            <h2>Student List</h2>

            <div>

                <Link to="/create">Create +</Link>

            </div>

            <table>

                <thead>

                    <tr>

                        <th>Student\_ID</th>

                        <th>Name</th>

                        <th>Age</th>

                        <th>Grade</th>

                        <th>Action</th>

                    </tr>

                </thead>

                <tbody>

                    {data.map((student) => (

                        <tr key={student.id}>

                            <td>{student.id}</td>

                            <td>{student.name}</td>

                            <td>{student.age}</td>

                            <td>{student.grade}</td>

                            <td>

                                <Link to={`/read/${student.id}`}><button>Read</button></Link>

                                <Link to={`/edit/${student.id}`}><button>Edit</button></Link>

                                <button onClick={() => handleDelete(student.id)}>Delete</button>

                            </td>

                        </tr>

                    ))}

                </tbody>

            </table>

        </div>

    );

}

export default Home;

**Read.jsx :-**

import React, { useEffect, useState } from 'react';

import { useParams, Link } from 'react-router-dom';

import axios from 'axios';

function Read() {

    const { id } = useParams();

    const [student, setStudent] = useState(null);

    useEffect(() => {

        axios.get(`http://localhost:8081/read/${id}`)

            .then(res => {

                console.log("API Response:", res.data);

                if (res.data.length > 0) {

                    setStudent(res.data[0]);

                } else {

                    console.log("Student not found");

                    setStudent(null);

                }

            })

            .catch(err => console.log("API Error:", err));

    }, [id]);

    return (

        <div>

            <h2>Student Detail</h2>

            {student ? (

                <>

                    <h2>ID: {student.id}</h2>

                    <h2>Name: {student.name}</h2>

                    <h2>Age: {student.age}</h2>

                    <h2>Grade: {student.grade}</h2>

                    <Link to="/">Back</Link>

                    <Link to={`/edit/${student.id}`}>Edit</Link>

                </>

            ) : (

                <h2>Loading...</h2>

            )}

        </div>

    );

}

export default Read;

**Update.jsx :-**

import React, { useEffect, useState } from 'react';

import { useParams, useNavigate } from 'react-router-dom';

import axios from 'axios';

function Update() {

    const { id } = useParams();

    const navigate = useNavigate();

    const [student, setStudent] = useState({});

    useEffect(() => {

        axios.get(`http://localhost:8081/read/${id}`)

            .then(res => {

                console.log("API Response:", res.data);

                if (res.data.length > 0) {

                    setStudent(res.data[0]);

                } else {

                    console.log("Student not found");

                    setStudent(null);

                }

            })

            .catch(err => console.log("API Error:", err));

    }, [id]);

    const [values, setValues] = useState({

        name: '',

        age: '',

        grade: ''

    });

    useEffect(() => {

        if (student) {

            setValues({

                name: student.name || '',

                age: student.age || '',

                grade: student.grade || ''

            });

        }

    }, [student]);

    const handleSubmit = (e) => {

        e.preventDefault();

        axios.post(`http://localhost:8081/update/${id}`, values)

            .then(res => {

                console.log("Update Successful:", res.data);

                navigate('/');

            })

            .catch(err => console.log("Update Error:", err));

    };

    return (

        <div className="d-flex vh-100 justify-content-center align-items-center bg-light">

            <div className="w-50 bg-white p-4 rounded shadow">

                <h2 className="text-center">Update Student</h2>

                <form onSubmit={handleSubmit}>

                    <div className="mb-3">

                        <label className="form-label">Name</label>

                        <input

                            type="text"

                            placeholder="Enter Name"

                            className="form-control"

                            value={values.name}

                            onChange={e => setValues({...values, name: e.target.value})}

                        />

                    </div>

                    <div className="mb-3">

                        <label className="form-label">Age</label>

                        <input

                            type="text"

                            placeholder="Enter Age"

                            className="form-control"

                            value={values.age}

                            onChange={e => setValues({...values, age: e.target.value})}

                        />

                    </div>

                    <div className="mb-3">

                        <label className="form-label">Grade</label>

                        <input

                            type="text"

                            placeholder="Enter Grade"

                            className="form-control"

                            value={values.grade}

                            onChange={e => setValues({...values, grade: e.target.value})}

                        />

                    </div>

                    <button type="submit" className="btn btn-primary w-100">Update</button>

                </form>

            </div>

        </div>

    );

}

export default Update;

**Server API :-**

**server.js :-**

import express from 'express'

import mysql from 'mysql'

import cors from 'cors'

const app = express();

app.use(cors());

app.use(express.json())

const db = mysql.createConnection({

    host: "localhost",

    port: "3306",

    user: "root",

    password: "",

    database: "mu\_2425\_test"

} , function(err) {

    if (err) {

        console.error("MySQL connection error:", err);

        return;

    }

    console.log("Connected to MySQL database.");

});

app.get('/', (req, res) => {

    const sql = "SELECT \* FROM students";

    db.query(sql, (err, result) => {

        if (err) {

            console.error("Database error:", err);

            return res.status(500).json({ message: "Error inside server", error: err });

        }

        return res.json(result);

    });

});

app.post('/student', (req, res) => {

    const sql = "INSERT INTO students (name, age, grade) VALUES (?, ?, ?)";

    const values = [req.body.name, req.body.age, req.body.grade];

    db.query(sql, values, (err, result) => {

        if (err) {

            console.error("MySQL Error:", err);

            return res.status(500).json({ error: err.message });

        }

        return res.json({ message: "Student added successfully", result });

    });

});

app.get('/read/:id', (req, res) => {

    const sql = "SELECT \* FROM students WHERE id=?";

    const id = req.params.id;

    db.query(sql, [id], (err, result) => {

        if (err) {

            console.error("Database error:", err);

            return res.status(500).json({ message: "Error inside server", error: err });

        }

        return res.json(result);

    });

});

app.post('/update/:id', (req, res) => {

    const sql = "UPDATE students SET name=?, age=?, grade=? WHERE id=?";

    const id = req.params.id;

    const { name, age, grade } = req.body;

    db.query(sql, [name, age, grade, id], (err, result) => {

        if (err) {

            console.error("Database error:", err);

            return res.status(500).json({ message: "Error inside server", error: err });

        }

        return res.json({ message: "Student updated successfully" });

    });

});

app.post('/delete/:id', (req, res) => {

    const sql = "DELETE FROM students WHERE id=?";

    const id = req.params.id;

    db.query(sql, [id], (err, result) => {

        if (err) {

            console.error("Error deleting student:", err);

            return res.status(500).json({ message: "Error deleting student", error: err });

        }

        return res.json({ message: "Student deleted successfully", result });

    });

});

app.listen(8081, () => {

    console.log("Listening");

})

**Output :-**





