Experiment 1

Name : Syed Adnan Hossain
UID : 23BCS11343
Branch: BE-CSE
Section/Group: KRG-3B
Semester: 6th
Date of Performance: 12-01-2026
Subject Name: Full Stack-II
Subject Code: 23CSH-309

1. Aim:
To design and implement the foundational frontend architecture of the EcoTrack application using modern React practices, Vite tooling, and ES6+ JavaScript features.

2. Objective:-

- To understand about basic of React and Vite.
- To create a project using Vite with proper flow.
- To apply ES6 array methods (map, filter, reduce) for data-driven UI rendering
- To separate concerns using components, pages, and data modules

3. Implementation/Code:

- **Logs.Js :-**

```
export const logs = [
  { id: 1, activity: "Car Travel", carbon: 12 },
  { id: 2, activity: "Electricity Usage", carbon: 5 },
  { id: 3, activity: "Cycling", carbon: 0 },
  { id: 4, activity: "Public Bus", carbon: 2 },
  { id: 5, activity: "Flight (Domestic)", carbon: 150 },
  { id: 6, activity: "Vegetarian Meal", carbon: 1 },
  { id: 7, activity: "Cycling", carbon: 0 },
  { id: 8, activity: "Electronics Recycling", carbon: -2 },
  { id: 9, activity: "Car Travel", carbon: 8 },
  { id: 10, activity: "Electricity Usage", carbon: 10 },
  { id: 11, activity: "Walking", carbon: 0 },
  { id: 12, activity: "Heating Usage", carbon: 15 },
  { id: 13, activity: "Train Journey", carbon: 3 },
  { id: 14, activity: "Meat Meal", carbon: 5 },
  { id: 15, activity: "Turned Off Lights", carbon: -1 },
]
```

- **Dashboard.Jsx :-**

```
import { logs } from '../data/log';

const Dashboard = () => {
  const totalCarbon = logs.reduce((sum, log) => {
    return sum + log.carbon;
  }, 0);
```

```
  const getNonZero=(logs)=>{
    return logs.filter(log=>log.carbon!==0)
  }
  const getCarbonColor = (carbon)=>{
    return carbon >= 4 ? "text-red-600" : "text-green-600";
  }

  const top2Logs = logs
    .slice()
    .sort((a, b) => b.carbon - a.carbon)
    .slice(0, 2);

  return (
    <div>
      <h1>Logs</h1>

      <div>Sum: {totalCarbon}</div>

      <h2>All Logs</h2>

      <ul>
        {getNonZero(logs).map((log) => (
          <li key={log.id}>
            <span className={`font-semibold ${getCarbonColor(log.carbon)}`}
>{log.activity}</span>
            <span className={`font-semibold ${getCarbonColor(log.carbon)}`}>
              {log.carbon} kg
            </span>
          </li>
        ))}
      </ul>


      <h2>Top 2 Carbon Emissions</h2>
      <ul>
        {top2Logs.map((log) => (
          <li key={log.id}>
            {log.activity} - {log.carbon} kg
          </li>
        ))}
      </ul>

      <h2></h2>
    </div>
  );
};
```

```
export default Dashboard;
```

- **Logs.Jsx :-**

```
import React from 'react'
import { logs } from '../data/log'

export const Logs = () => {

  const highCarbon = logs.filter(
    log => log.carbon >= 4
  )

  return (
    <div>
      <h2>High Carbon Activities more than 4</h2>

      {highCarbon.map((log, index) => (
        <p key={index}>
          {log.activity} - {log.carbon} kg
        </p>
      ))}
    </div>
  )
}

export const LowCarbon = () => {
  const LowCarbon = logs.filter(
    logs =>logs.carbon <=3
  )
  return (
    <div>
      <h2>Low Carbon Activities less than 3</h2>

      {logs.map((log,index)=>{
        return (
          <p>
            {log.activity} - {log.carbon} kg
          </p>
        )
      })}
    </div>
  )
}
```

- **App.Jsx:-**

```
import React from 'react'
import { useState } from 'react'
import Dashboard from './pages/dashboard'
import { Logs , LowCarbon} from './pages/logs'
import Header from './components/Header'

function App() {
  const [count, setCount] = useState(0)

  return (
    <div>
        <Header title="data"/>
        <Dashboard  />
        <Logs />
        <LowCarbon />
    </div>
  )
}

export default App
```

4. Output

5. Learning Outcome :-

- Developed an **Eco Tracker application** using React to analyze carbon emissions.
- Implemented a **Dashboard component** to display emission data from JavaScript logs.
- Used **map(), filter(), and reduce()** to classify low and high carbon emission elements.
- Applied **component-based architecture** for better code organization.
- Achieved **dynamic data rendering** based on emission levels.
- Enhanced understanding of **data processing and state-driven UI in React.**