**WEEK 7**

**Module 10 – Single Page Application framework – React**

**14. ReactJS-HOL**

**CODE:**

**src/ThemeContext.js**

import { createContext } from 'react';

const ThemeContext = createContext('light');

export default ThemeContext;

**src/App.js**

import React, { useState } from 'react';

import EmployeesList from './components/EmployeesList';

import ThemeContext from './ThemeContext';

function App() {

const [theme, setTheme] = useState('light');

const toggleTheme = () => {

setTheme(prev => (prev === 'light' ? 'dark' : 'light'));

};

return (

<ThemeContext.Provider value={theme}>

<div className={app ${theme}}>

<h1>Employee Management</h1>

<button onClick={toggleTheme}>Toggle Theme</button>

<EmployeesList />

</div>

</ThemeContext.Provider>

);

}

export default App;

**src/components/EmployeesList.js**

import React from 'react';

import EmployeeCard from './EmployeeCard';

const EmployeesList = () => {

const employees = [

{ id: 1, name: 'Darwin', role: 'Developer' },

{ id: 2, name: 'Sophia', role: 'Designer' }

];

return (

<div>

{employees.map(emp => (

<EmployeeCard key={emp.id} employee={emp} />

))}

</div>

);

};

export default EmployeesList;

**src/components/EmployeeCard.js**

import React, { useContext } from 'react';

import ThemeContext from '../ThemeContext';

const EmployeeCard = ({ employee }) => {

const theme = useContext(ThemeContext);

return (

<div className={card ${theme}}>

<h3>{employee.name}</h3>

<p>{employee.role}</p>

<button className={theme}>View</button>

</div>

);

};

export default EmployeeCard;

**src/index.js**

import React from 'react';

import ReactDOM from 'react-dom/client';

import App from './App';

import './index.css';

const root = ReactDOM.createRoot(document.getElementById('root'));

root.render(<App />);

**index.css**

.card.light {

background: #fff;

color: #000;

}

.card.dark {

background: #333;

color: #fff;

}

button.light {

background: #eee;

color: #000;

}

button.dark {

background: #555;

color: #fff;

}

**OUTPUT:**



