**14. ReactJS-HOL**

Developers of Apps Centric Solutions have created an employee management application which supports light and dark themes for the buttons. The current solution uses the react state and props to provide the theme name to be used from App component to Employee List component and from there to Employee Card component. Quality assurance team analyzed the solutions and found the technique being used to be a substandard one. React architect suggested to use the react context API to share the theme name with nested child components instead of passing them down using props from the parent component.

**React App : employeesapp**

**App.js**

import './App.css';

import { EmployeesData } from './Employee';

import EmployeesList from './EmployeesList';

import { useState } from 'react';

import ThemeContext from './ThemeContext';

function App() {

  const Employees = EmployeesData;

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

  return (

    <>

      <div>

        <label>SELECT A THEME </label>

        <select onChange={(e) => setTheme(e.target.value)}>

          <option value='light'>Light</option>

          <option value='dark'>Dark</option>

        </select>

      </div>

      <ThemeContext.Provider value={theme}>

        <EmployeesList employees={Employees} />

      </ThemeContext.Provider>

    </>

  );

}

export default App;

**Employee.js**

class Employee {

    constructor(id, name, email, phone){

        this.id=id;

        this.name=name;

        this.email=email;

        this.phone=phone;

    }

}

const EmployeesData=[

    new Employee(101,'Jojo','jojo@congizant.com','98238971234'),

    new Employee(102,'Sam','sam@congizant.com', '9981184126'),

    new Employee(103,'Elisa','elisa@cognizant.com','9989389735')

];

export default Employee;

export {EmployeesData};

**EmployeeCard.js**

import Styles from './EmployeeCard.module.css';

import { useContext } from 'react';

import ThemeContext from './ThemeContext';

function EmployeeCard(props) {

  const theme = useContext(ThemeContext);

  return (

    <div className={`${Styles.Card} ${Styles[theme]}`}>

      <h3>{props.employee.name}</h3>

      <p>{props.employee.email}</p>

      <p>{props.employee.phone}</p>

      <p>

        <a href="#" className={theme}>Edit</a>

        <a href="#" className={theme}>Delete</a>

      </p>

    </div>

  );

}

export default EmployeeCard;

**EmployeeCard.module.css**

.Card {

  display: inline-block;

  width: 25%;

  border: 1px solid #999;

  border-radius: 5px;

  margin: 10px;

  padding: 10px;

}

.light {

  background-color: white;

  color: black;

}

.dark {

  background-color: black;

  color: white;

}

.light a {

  color: #007bff;

  margin-right: 10px;

  text-decoration: none;

}

.dark a {

  color: #00d8ff;

  margin-right: 10px;

  text-decoration: none;

}

**EmployeeList.js**

import EmployeeCard from './EmployeeCard';

function EmployeesList(props) {

  return (

    <div>

      <h1>Employees List</h1>

      {

        props.employees.map(employee =>

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

        )

      }

    </div>

  );

}

export default EmployeesList;

**Final Output:**



