**ReactJS Handson**

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.

You are assigned the task of converting the application form props only to React Context API.

Application can be downloaded from below



1. Unzip the application and open it using VS Code
2. Go to terminal and execute *npm install* command to restore all the node modules
3. Run the application once to see the output. Use npm start command.
4. Explore the components present in **App.js**, **EmployeesList.js** and **EmployeeCard.js** files.
5. Create a new file with the name as **ThemeContext.js**. Define a new context in the file with the name as ThemeContext and assign it a default value of ‘light’ and export it as default form the module.
6. Open App component present in **App.js** file.
   1. Import the ThemeContext in App component.
   2. Define the theme context provider to be the entire JSX of the App component.
   3. Assign the value for the theme provider from the state of the component.
   4. Modify the call to EmployeeList component so that theme name is no longer passed as props.
7. Go to EmployeeList component present in **EmployeeList.js** file and modify it so that theme name is not passed explicitly to its child component.
8. Go to **EmployeeCard** component inside **EmployeeCard.js** file
   1. Import the ThemeContext into the component file
   2. Retrieve the value of the context with the help of **useContext()** and store it in a variable
   3. Use the variable to pass the className for the buttons.

**ThemeContext.js**

import React from 'react';

const ThemeContext = React.createContext('light');

export default ThemeContext;

**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 (

    <>

    <ThemeContext.Provider value={theme}>

      <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>

      <EmployeesList employees={Employees} />

    </ThemeContext.Provider >

    </>

  );

}

export default App;

**EmployeesList.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;

**EmployeeCard.js**

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

import React, { useContext } from 'react';

import ThemeContext from './ThemeContext';

function EmployeeCard(props) {

    const theme = useContext(ThemeContext);

    return (

        <div className={Styles.Card}>

            <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>

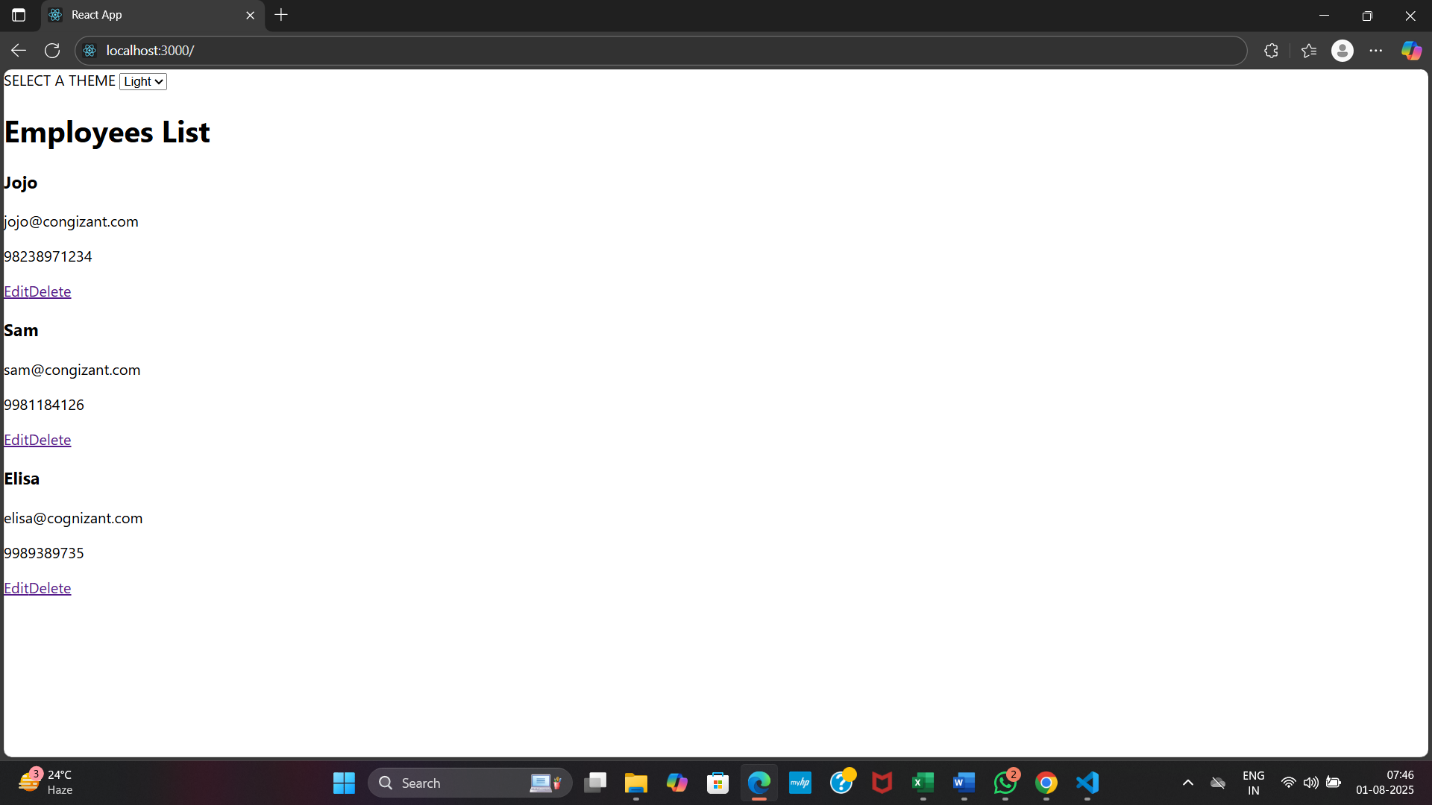
        </div>

    );

}

export default EmployeeCard;

**OUTPUT**

****