

14.ReactJS-HOL

```
import { createContext } from 'react';

const ThemeContext = createContext('light');
export default ThemeContext;

import React from 'react';
import EmployeeCard from './EmployeeCard';

function EmployeesList() {
  const employees = [
    { id: 1, name: 'Alice', role: 'Developer' },
    { id: 2, name: 'Bob', role: 'Designer' },
    { id: 3, name: 'Charlie', role: 'Tester' }
  ];

  return (
    <div>
      {employees.map(emp => (
        <EmployeeCard key={emp.id} employee={emp} />
      ))}
    </div>
  );
}

export default EmployeesList;

import React, { useContext } from 'react';
import ThemeContext from './ThemeContext';
```

```

function EmployeeCard({ employee }) {
  const theme = useContext(ThemeContext);

  return (
    <div className="employee-card">
      <h3>{employee.name}</h3>
      <p>{employee.role}</p>
      <button className={theme === 'dark' ? 'dark-btn' : 'light-btn'}>
        View Details
      </button>
    </div>
  );
}

```

```

export default EmployeeCard;

import React, { useState } from 'react';
import EmployeesList from './EmployeesList';
import ThemeContext from './ThemeContext';
import './App.css';

```

```

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

  const toggleTheme = () => {
    setTheme(prevTheme => (prevTheme === 'light' ? 'dark' : 'light'));
  };

  return (

```

```
<ThemeContext.Provider value={theme}>
  <div className="App">
    <h1>Employee Management App</h1>
    <button onClick={toggleTheme}>
      Toggle Theme ({theme})
    </button>
    <EmployeesList />
  </div>
</ThemeContext.Provider>
);
}
```

```
export default App;

.App {
  font-family: Arial, sans-serif;
  padding: 20px;
}
```

```
.employee-card {
  margin: 10px 0;
  padding: 10px;
  border: 1px solid #ccc;
}
```

```
.light-btn {
  background-color: white;
  color: black;
  border: 1px solid black;
```

```
padding: 5px 10px;  
}
```

```
.dark-btn {  
  background-color: black;  
  color: white;  
  border: 1px solid white;  
  padding: 5px 10px;  
}
```

OUTPUT:

