**WEEK – 6(Additional HandsOn)**

**React**

**7. ReactJS-HOL**

**Define Props:**

Props (short for *properties*) are read-only objects passed from a parent React component to a child component. They allow parent components to configure or communicate data and event handlers to child components. Props let React components be reusable and customizable by passing different values:

<MyButton color="blue" label="Click Me" />

Here, color and label are props received by the MyButton component.

**Explain Default Props:**

Default Props are property values assigned to a React component if the parent doesn't provide those props. This ensures the component still works as expected even if certain props are omitted:

function Greeting({ name }) {

return <h1>Hello, {name}!</h1>; }

Greeting.defaultProps = { name: "Guest",

};

If the name prop isn't provided, it defaults to "Guest".

In function components with ES6, you can also use default parameters:

function Greeting({ name = "Guest" }) {

return <h1>Hello, {name}!</h1>;

}

**Identify the differences between State and Props:**

|  | Props State |
| --- | --- |
| Definition | External data passed to a component by its parent | Internal data managed within the component |
| Mutability | Read-only (immutable from the component’s perspective) | Mutable (can be changed by the component itself) |
| Who sets? | Set by parent (or React itself) | Set and managed by the component |
| Usage | Customize and configure how a component behaves or displays | Store variable data that changes during component’s lifecycle (like user input, toggles, etc.) |
| Access | Accessed via props object | Accessed via state object and managed with setState (class) or useState (function) |
| Update | Cannot be updated by the receiving component | Updated using this.setState (class) or useState hook (function) |

**Explain reactDOM.render():**

ReactDOM.render() is a method used to render a React element or component hierarchy into a DOM node in the browser. It’s how you instruct React to draw your UI on the page:

import React from 'react';

import ReactDOM from 'react-dom';

ReactDOM.render(<App />, document.getElementById('root'));

**CODE:**

**src/OnlineShopping.js:**

import React, { Component } from 'react';

import Cart from './Cart';

class OnlineShopping extends Component {

constructor(props) {

super(props);

this.items = [

{ itemname: 'Laptop', price: 80000 },

{ itemname: 'TV', price: 120000 },

{ itemname: 'Washing Machine', price: 50000 },

{ itemname: 'Mobile', price: 30000 },

{ itemname: 'Fridge', price: 70000 }

];

}

render() {

return (

<div style={{ textAlign: 'center', marginTop: '30px' }}>

<h2 style={{ color: 'green' }}>Items Ordered :</h2>

<table

style={{

margin: '0 auto',

borderCollapse: 'collapse',

width: '400px',

color: 'green' // ✅ all table text in green

}}

border="1"

>

<thead>

<tr>

<th>Name</th>

<th>Price</th>

</tr>

</thead>

<tbody>

{this.items.map((item, index) => (

<Cart key={index} itemname={item.itemname} price={item.price} />

))}

</tbody>

</table>

</div>

);

}

}

export default OnlineShopping;

**src/Cart.js:**

import React, { Component } from 'react';

class Cart extends Component {

render() {

return (

<tr>

<td>{this.props.itemname}</td>

<td>{this.props.price}</td>

</tr>

);

}

}

export default Cart;

**src/App.js:**

import React from 'react';

import './App.css';

import OnlineShopping from './OnlineShopping';

function App() {

return (

<div className="App">

<OnlineShopping />

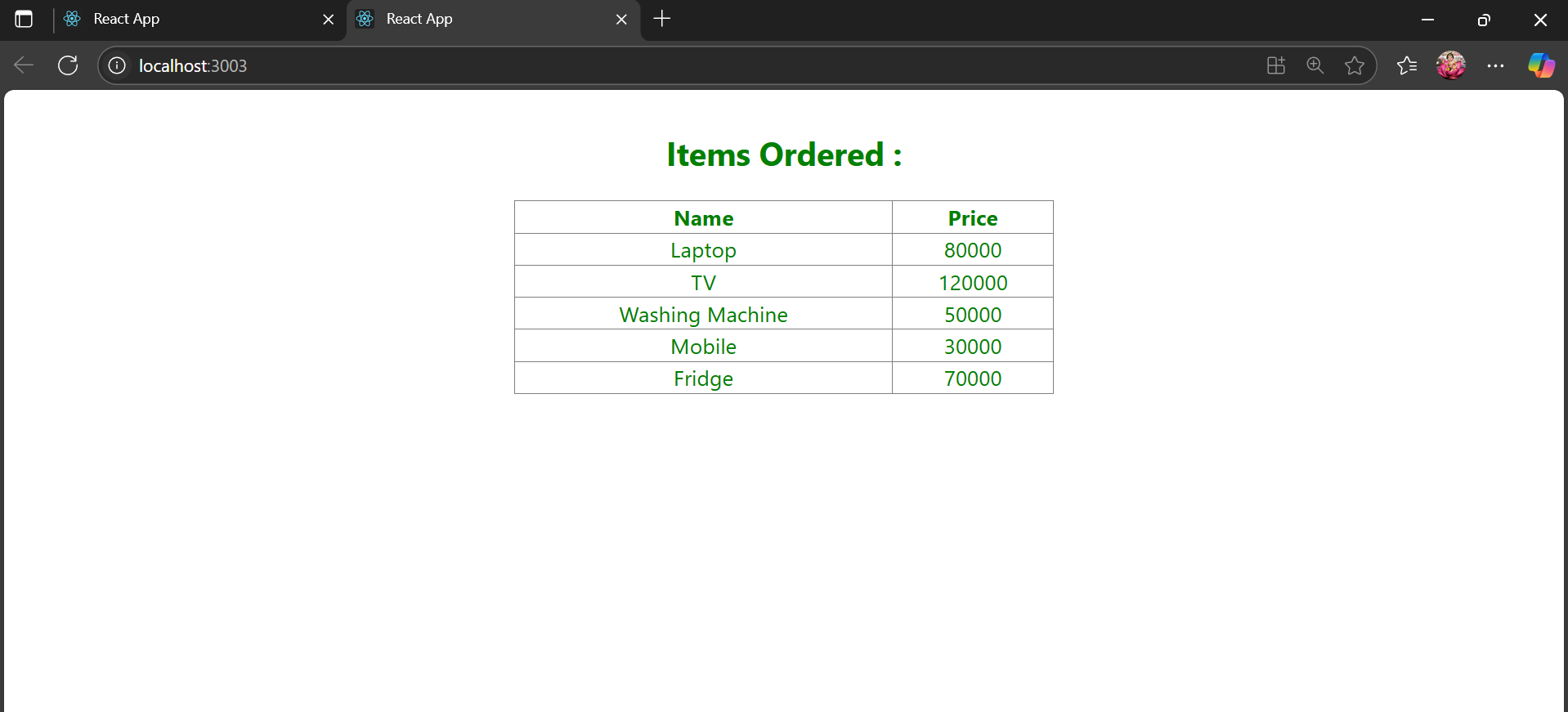
</div>

);

}

export default App;

**OUTPUT:**

****