

Cheat Sheet: Introduction to React and Class Components

JSX syntax and Class Component	Description	Code Example
Vite tool	This tool is used to create React projects with the latest version	<pre>npm create vite@latest</pre>
JSX syntax with fragments	Jsx allows to include multiple tags within single parent tag only. For this you can include empty tags also known as fragments <code><></></code>	<pre><> <h1>This is a heading tag</h1> </></pre>
JSX syntax with parent tag	Jsx allows to include multiple tags within single parent tag only. For this you can include tags like div tag	<pre><div> <h1>This is a heading tag</h1> </div></pre>
Class component	A class component extends the Component class with a render method	<pre>import React, { Component } from 'react' export default class Extra extends Component { render() { return (<> <p>This is class component</p> </>) } }</pre>
State management in class component	State can be managed using <code>this.state</code> within constructor of class component	<pre>import React, { Component } from 'react' export default class Extra extends Component { constructor(){ super(); this.state={ count:0 } } render() { return (<> <p>This is paragraph</p> </>) } }</pre>

Access data of state management within class component	<p>Data can be accessed by including variable name declared within {} and include variable name within these curly braces</p>	<pre>import React, { Component } from 'react' export default class Extra extends Component { constructor(){ super(); this.state={ count:0 } } render() { return (<> <p>the count is {this.state.count}</p> </>) } }</pre>
Props in class component	<p>Props can be sent from parent component as attribute along with child component</p>	<pre>import React, { Component } from 'react' import ChildComponent from './ChildComponent' export default class App extends Component { constructor(){ super() this.state={ title:"Manager" } } render() { return (<> <Extra title={this.state.title}/> </>) } }</pre>
Access props within child class component	<p>Props can be accessed easily within the child class component using this.props.variable_name</p>	<pre>import React, { Component } from 'react' export default class Extra extends Component { constructor(props){ super(); this.state={ count:0 } } render() { return (<> <p>The count value is {this.props.title}</p> </>) } }</pre>
Event handling in class component	<p>Events such as click events are handled by defining a function before the render method</p>	<pre>import React, { Component } from 'react'; class MyComponent extends Component { handleClick() { alert('Button clicked!'); } render() { return (<div> <h1>Event Handling in Class Component</h1> <button onClick={this.handleClick}>Click Me</button> </div>); } } export default MyComponent;</pre>



Skills Networ