

Requirements

- NPM - It is used to take advantage of a vast ecosystem of third-party packages, and easily install or update them.
- Webpack - webpack is a static module bundler for modern JavaScript applications. When webpack processes your application, it internally builds a dependency graph which maps every module your project needs and generates one or more bundles.
- Babel - Babel is a toolchain that is mainly used to convert ECMAScript 2015+ code into a backwards compatible version of JavaScript in current and older browsers or environments.

React.createElement(type, props, children)

React.createElement(type, props, children) - It creates a React Element with the given arguments.

Syntax:- React.createElement(type, props, children)

- type: Type of the html element or component. (example : h1,h2,p,button..etc).
- props: The properties object.
Example: {style :{ color:"blue"}} or className or event handlers etc.
- children: anything you need to pass between the dom elements.

Ex:-

```
React.createElement('div', null, 'Hello World');
```

ReactDOM.render(element, DOMnode)

ReactDOM.render(element, DOMnode) - It takes a React Element and render it to a DOM node.

Syntax:- ReactDOM.render(element, DOMnode)

- The first argument is which component or element needs to render in the dom.
- The second argument is where to render in the dom.

Ex:-

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

render() Method

The render() method is the only required method in a class component. It examines this.props and this.state . It returns one of the following types:

React elements – These are created via JSX(Not required).

For example, <div /> and <App /> are React elements that instruct React to render a DOM node, or another user-defined component, respectively.

Arrays and fragments - It is used to return multiple elements from render.

Portals – it is used to render children into a different DOM subtree.

String and numbers - These are rendered as text nodes in the DOM.

Booleans or null - It renders nothing. (Mostly exists to support return test && <Child /> pattern, where test is boolean.)

Note - The render() function should be pure, meaning that it does not modify component state, it returns the same result each time it's invoked, and it does not directly interact with the browser.