



03

createElement & createRoot

Our Target

Hello React!!



The screenshot shows a web browser window with the text "Hello React!!" displayed. Below the browser window, the developer tools are open to the "Elements" tab. The HTML structure is as follows:

```
<meta charset="UTF-8">
<title>Hello React</title>
</head>
<body>
  <!-- step1: create a root div -->
  <div id="root"> == $0
    <div>Hello React!!</div>
  </div>
  <script>...</script>
  <!-- Code injected by live-server -->
  <script>...</script>
</body>
```

A red arrow points from the "Hello React!!" text in the browser window to the inner `<div>Hello React!!</div>` element in the developer tools. A red box highlights the `<div id="root">` element and its children.

Our Target

Hello React!!



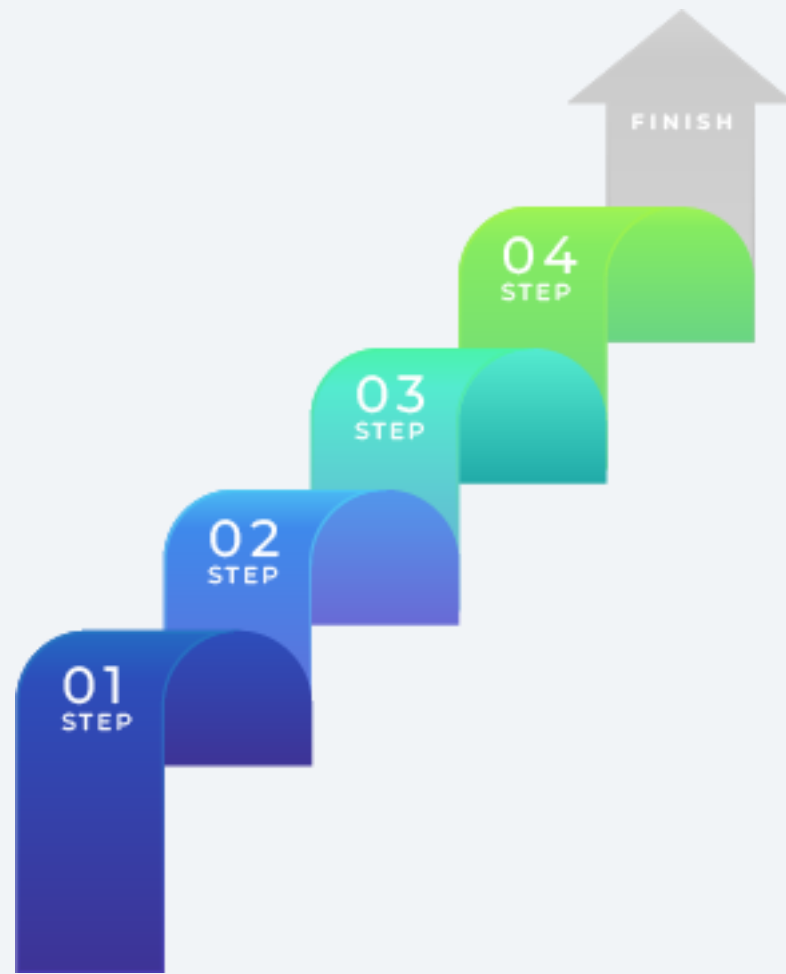
The screenshot shows a web browser window with the text "Hello React!!" displayed. Below the browser window, the developer tools are open to the "Elements" tab. The HTML structure is as follows:

```
<meta charset="UTF-8">
<title>Hello React</title>
</head>
<body>
  <!-- step1: create a root div -->
  <div id="root"> == $0
    <div>Hello React!!</div>
  </div>
  <script>...</script>
  <!-- Code injected by live-server -->
  <script>...</script>
</body>
```

A red arrow points from the "Hello React!!" text in the browser window to the inner `<div>Hello React!!</div>` element in the developer tools. A red box highlights the `<div id="root">` element and its contents.

JS **X**





Import React & ReactDOMClient

```
<script src=  
  "https://ajax.googleapis.com/ajax/libs/jquery/3.7.  
  1/jquery.min.js"  
></script>
```

Import React & ReactDOMClient

👉 React: for creating React Virtual DOM

👉 ReactDOMClient: for rendering the UI in the browser

Import React & ReactDOMClient

```
<script src=  
  "https://unpkg.com/react@18/umd/react.production.min.js"  
></script>  
  <script src=  
    "https://unpkg.com/react-dom@18/umd/react-dom.production.min.js"  
  ></script>
```

UMD Way of Using React



Import React & ReactDOMClient

```
<script type="module">
  import React from "https://esm.sh/react@19/?dev";
  import ReactDOMClient from "https://esm.sh/react-dom@19/client?dev";
</script>
```

In React 19, you can use modern ESM module format

Two Raw React APIs

👉 `React.createElement`

👉 `ReactDOMClient.createRoot`

Two Raw React APIs

👉 React.createElement

```
const element = createElement(type, props, ...children);
```

Parameters:

type: valid React component type: tag name string (such as 'div' or 'span'), or a React component (a function, a class, or a special component like Fragment)

props: must either be an object or null. If you pass null, it will be treated the same as an empty object

optional ...children: Zero or more child nodes. They can be any React nodes.

Two Raw React APIs

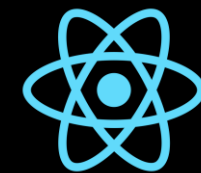
👉 ReactDOMClient.createRoot

createRoot lets you create a root to display React components inside a browser DOM node.

It creates a React root instance that serves as the entry point for managing and rendering React components into the real DOM.

Call `render` to render a React component inside of it:

```
React.createRoot(domNode).render(reactNode);
```



Thanks for watching