

import ReactDOM from 'react-dom/client'

The statement `import ReactDOM from 'react-dom/client'` is used in React applications, specifically starting from React 18, to import the client-side rendering capabilities of ReactDOM.

Explanation:

- `ReactDOM`: This refers to the core React package that provides methods for interacting with the Document Object Model (DOM). It's responsible for rendering React components into the browser's DOM.
- `'react-dom/client'`: This specific import path indicates that you are importing the client-specific methods of ReactDOM. In earlier versions of React (prior to React 18), you would typically import `ReactDOM` directly from `'react-dom'`. However, with the introduction of React 18 and its concurrent features, the client-side rendering functionalities were separated into a dedicated `'react-dom/client'` module.

Key functions you would typically use after this import:

- `createRoot(container[, options])`: This function is used to create a React root for a given DOM element (`container`). It returns a root object that you can then use to render your React application.

```
import ReactDOM from 'react-dom/client';

const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
  // Your React components here
);
```

- `hydrateRoot(container, element[, options])`: This function is used for hydrating server-rendered HTML. If you are using server-side rendering (SSR), `hydrateRoot` is used on the client to attach event listeners and make the server-rendered HTML interactive.

In essence, `import ReactDOM from 'react-dom/client'` is the modern way to set up client-side rendering in React 18 and later, providing access to the `createRoot` and `hydrateRoot` functions for managing your React application's lifecycle in the browser

Revision #2

Created 29 October 2025 02:43:39 by AI API

Updated 9 December 2025 11:46:11 by AI Channel