

React DOM APIs

The `react-dom` package contains methods that are only supported for the web applications (which run in the browser DOM environment). They are not supported for React Native.

APIs

These APIs can be imported from your components. They are rarely used:

- `createPortal` lets you render child components in a different part of the DOM tree.
- `flushSync` lets you force React to flush a state update and update the DOM synchronously.

Resource Preloading APIs

These APIs can be used to make apps faster by pre-loading resources such as scripts, stylesheets, and fonts as soon as you know you need them, for example before navigating to another page where the resources will be used.

[React-based frameworks](#) frequently handle resource loading for you, so you might not have to call these APIs yourself. Consult your framework's documentation for details.

- `prefetchDNS` lets you prefetch the IP address of a DNS domain name that you expect to connect to.
- `preconnect` lets you connect to a server you expect to request resources from, even if you don't know what resources you'll need yet.
- `preload` lets you fetch a stylesheet, font, image, or external script that you expect to use.
- `preloadModule` lets you fetch an ESM module that you expect to use.
- `preinit` lets you fetch and evaluate an external script or fetch and insert a stylesheet.

- `preinitModule` lets you fetch and evaluate an ESM module.
-

Entry points

The `react-dom` package provides two additional entry points:

- `react-dom/client` contains APIs to render React components on the client (in the browser).
 - `react-dom/server` contains APIs to render React components on the server.
-

Removed APIs

These APIs were removed in React 19:

- `findDOMNode`: see [alternatives](#).
- `hydrate`: use `hydrateRoot` instead.
- `render`: use `createRoot` instead.
- `unmountComponentAtNode`: use `root.unmount()` instead.
- `renderToNodeStream`: use `react-dom/server` APIs instead.
- `renderToStaticNodeStream`: use `react-dom/server` APIs instead.

PREVIOUS

< `<title>`

NEXT

`createPortal` >

Managing State

Escape Hatches

Community

[Code of Conduct](#)

[Meet the Team](#)

[Docs Contributors](#)

[Acknowledgements](#)

More

[Blog](#)

[React Native](#)

[Privacy](#)

[Terms](#)

