01/10

React Lazy Loading

SWIPE

Lazy Loading

Lazy Loading is a programming concept mostly used in the context of loading resources, such as data, images, modules, or other elements, on a delayed or on-demand basis, rather than loading them immediately when a program is started.

This is done with the aim of improving the performance and efficiency of a program or application by avoiding unnecessary loading of resources that may not be used immediately.

ONE MORE

The basic idea of Lazy Loading is to postpone the loading of resources until they are actually needed. This can be especially useful in situations where a program has many features, but not all of them are used all the time.

ONE MORE

React Lazy Loading

In the React.js library, this technique is used to improve application performance by loading components only on demand, that is, only when they are really needed. And this is done using the resources: lazy (React.lazy() function) and Suspense (Component).

The lazy( ) function allows you to postpone component loading until it is rendered for the first time, for example, when a user clicks a button.

ONE MORE

That way:

```
import { lazy } from 'react';

const MyComponentLazy= lazy(() => import('./MyComponent.js'));
```

ONE MORE

Now that the component loads on demand, you need to specify what should be shown while it is loading. This can be done by wrapping the lazy component or any of its parents in a **<Suspense>** boundary.

That way:

```
<Suspense fallback={<Loading />}>
  <h2>My component lazy</h2>
  <MyComponentLazy />
 </Suspense>
```

According to the code above, the MyComponentLazy component won't be loaded until there is an attempt, usually by the user, to render it. And, while the MyComponentLazy hasn't been loaded yet, the Loading component will be shown in its place.

ONE MORE

React.js Router Lazy Loading

What has been shown so far is Component Lazy, but route Lazy is also common, using a routing library like React Router.

```jsx
import React, { lazy, Suspense } from 'react';
import { BrowserRouter as Router, Route, Switch } from 'react-router-dom';

const HomePage = lazy(() => import('./HomePage'));
const AboutPage = lazy(() => import('./AboutPage'));
const ContactPage = lazy(() => import('./ContactPage'));

function App() {
  return (
    <Router>
      <Suspense fallback={<div>Loading...</div>}>
        <Switch>
          <Route path="/" exact component={HomePage} />
          <Route path="/about" component={AboutPage} />
          <Route path="/contact" component={ContactPage} />
        </Switch>
      </Suspense>
    </Router>
  );

export default App;
```

ONE MORE

In this example, each page (**HomePage**, **AboutPage**, **ContactPage**) is loaded on demand when the corresponding route is accessed, rather than all at once.

Lazy Loading is a powerful technique for improving performance and reducing initial load size in React applications, allowing you to load components or resources only when they are needed. This can be especially useful in large, complex applications.

ONE MORE

10/10

What do you think about Lazy Loading in the React.js?

PABLO CAVALCANTE

COMMENT BELOW