Assignment 9

1. When and why do we need lazy()?

The React.lazy function lets you render a dynamic import as a regular component.

Before:

jsx

```
import OtherComponent from './OtherComponent';
```

After:

jsx

```
const OtherComponent = React.lazy(() => import('./OtherComponent'));
```

This will automatically load the bundle containing the OtherComponent when this component is first rendered.

React.lazy takes a function that must call a dynamic import(). This must return a Promise which resolves to a
module with a default export containing a React component.

2. What is suspense?

The lazy component should then be rendered inside a Suspense component, which allows us to show some fallback
content (such as a loading indicator) while we're waiting for the lazy component to load.

jsx

```
import React, { Suspense } from 'react';

const OtherComponent = React.lazy(() => import('./OtherComponent'));

function MyComponent() {
  return (
    <div>
      <Suspense fallback={<div>Loading...</div>}>
        <OtherComponent />
      </Suspense>
    </div>
  );
}
```

The fallback prop accepts any React elements that you want to render while waiting for the component to load. You can
place the Suspense component anywhere above the lazy component. You can even wrap multiple lazy components with a
single Suspense component.

3. Why we got this error : A component suspended while responding to synchronous input. This will cause the UI to be replaced with a loading indicator. To fix, updates that suspend should be wrapped with startTransition?How does suspense fix this error?

There are sometimes when component under suspense takes sometime to render, but we also don't want to show loader
or the fallback component then we can tell react to show the old component only and render the new one when its ready
using startTransition method.

Transitions are a new concurrent feature introduced in React 18. They allow you to mark updates as transitions,
which tells React that they can be interrupted and avoid going back to Suspense fallbacks for already visible content.
jsx
React.startTransition(callback)

4. Advantages or disadvantages of code splitting?

code splitting can be very helpful in terms of improving the performance as the different bundles can be loaded when required.
The only drawback about code splitting is that you can only use it in client side rendering. Both the above techniques
won't work during server side rendering when it comes to React. This is the reason why the React team recommends using
loadable components for code splitting in the server.

5.when do we and why do we need suspense?

In essence, Suspense is a mechanism for React developers to **indicate towards React that a component is waiting for data to be ready**. React then knows that it should wait for that data to be fetched. In the meanwhile, a fallback will be shown to the user and React will continue with rendering the rest of the application.