

# REACT - DĚLENÍ KÓDU



Většina React aplikací bude mít svůj kód zabalený pomocí nástrojů jako je Webpack, Rollup, nebo Browserify.

## BUNDLING

Bundling je proces následování importnutých souborů a jejich slučování do jednoho souboru. K bundlování můžeme využít třeba Webpack. Pokud používáme nástroj jako je Create React App, tak je pro nás Webpack již nastavený a nemusíme si jej konfigurovat sami.

## DYNAMIC IMPORTING

Bundling je super, ale jak aplikace roste, tak se zvyšuje načítání. Kvůli tomu nemusíme všechny kód slučovat do jednoho souboru, ale můžeme některý načíst pomocí dynamického importu. V Reactu máme k dispozici funkci `React.lazy`, která nám umožňuje vyrenderovat dynamický import jako komponentu. Tato funkce bere jako parametr funkci, která musí zavolat dynamický import. Tato funkce se zavolá při prvním vyrenderování komponenty.

Lazy komponenty by se měli renderovat uvnitř komponenty `Suspense`, která nám umožňuje nějak zobrazit že se lazy komponenty načítají.

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

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

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

Pokud chceme nějak ošetřit, když se module nepodaří z nějakého důvodu načíst, tak to můžeme provést pomocí `Error Boundaries`. Po vytvoření naší `Error Boundary` ji můžeme použít kdekoliv nad komponentou `Suspense`.

```
<MyErrorBoundary>
  <Suspense fallback={<div>Loading...</div>}>
    <OtherComponent />
  </Suspense>
</MyErrorBoundary>
```