

LECTURE #5

By: Aqib Rehman

OUTLINE

Recap & Virtual DOM

The React Router

Links & NavLink

Programmatic Redirects

Higher Order Components

Using Axios

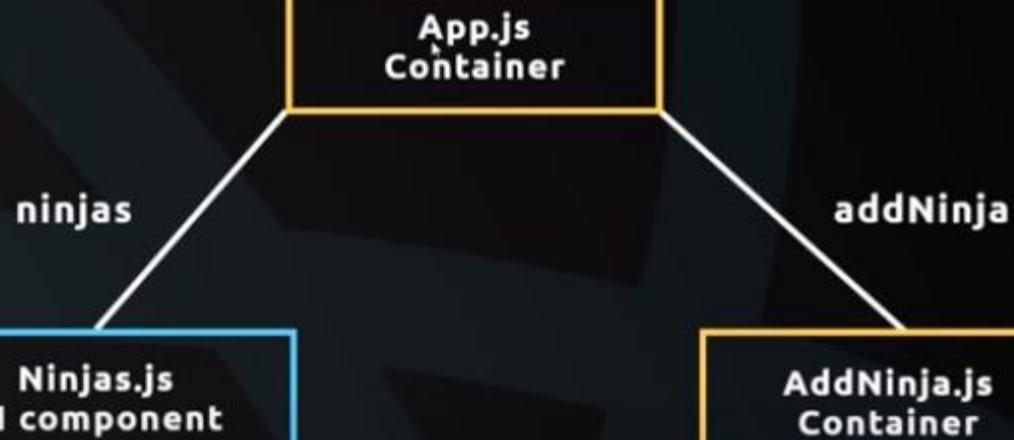
Route Parameters

Switch Tag

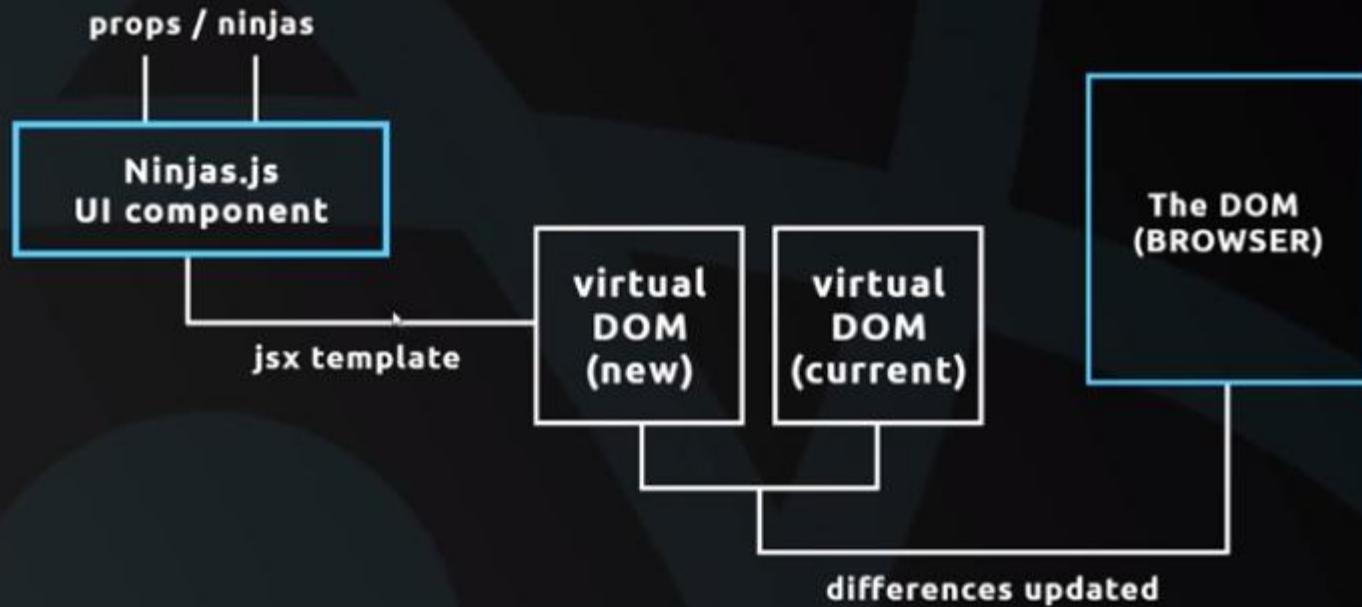
Importing Images

RECAP & VIRTUAL DOM

Recap & Virtual DOM

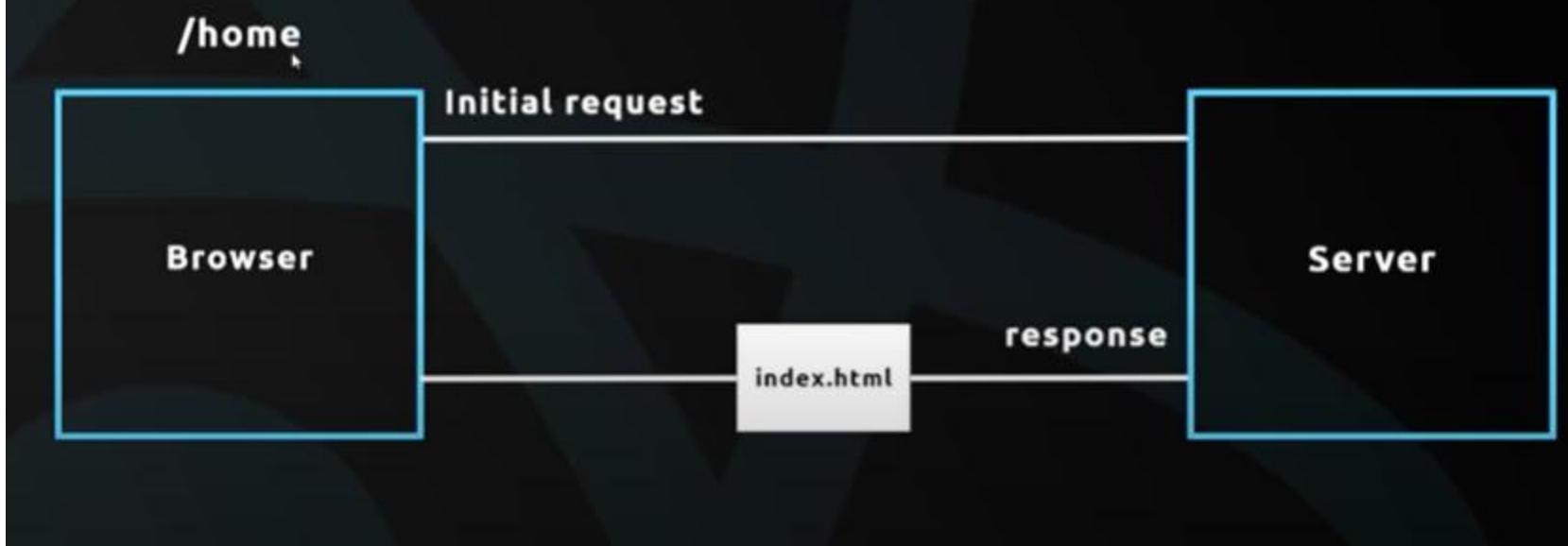


Recap & Virtual DOM



THE REACT ROUTER

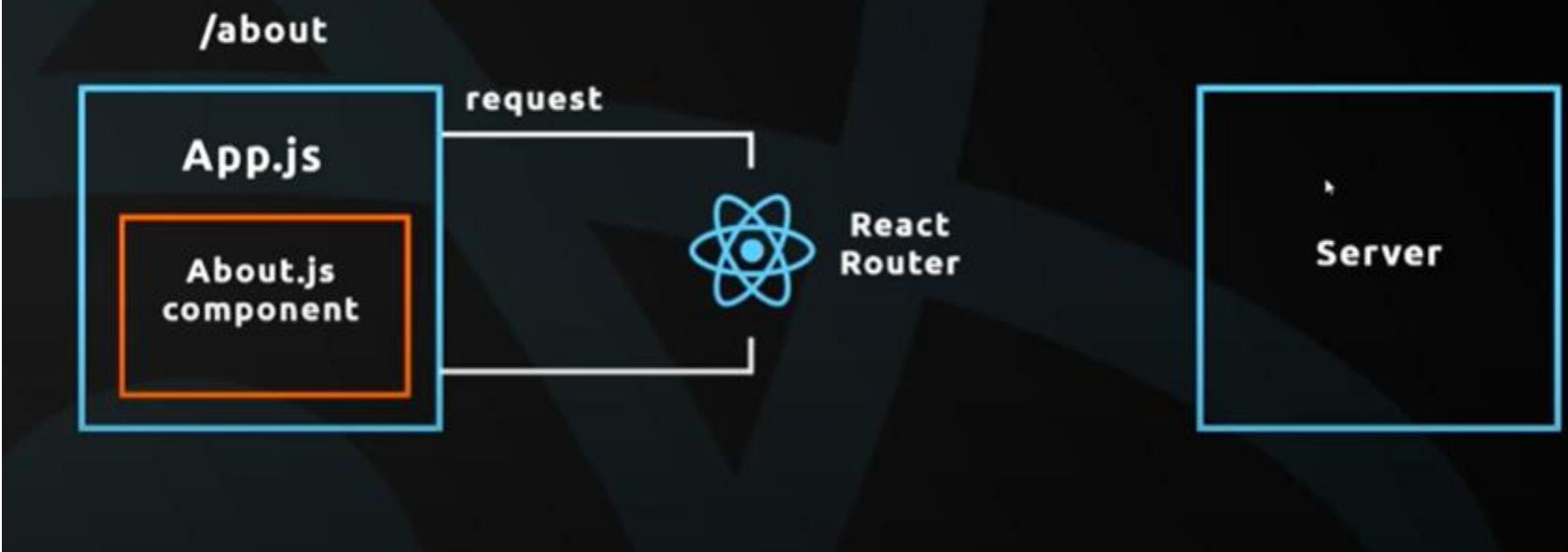
React Router



React Router



React Router



A screenshot of the Visual Studio Code interface. The left sidebar shows the file structure of a React application named 'poketimes'. The 'src' folder contains 'App.js' and 'components'. Inside 'components', there is a file named 'Home.js'. The main editor area displays the code for 'Home.js':

```
1 import React from 'react'
2
3 const Home = () => {
4   return (
5     <div className="container">
6       <h4 className="center">Home</h4>
7       <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adip
8     </div>
9   )
10 }
11
12 export default Home
```

The cursor is positioned at the end of the 'Home' export statement. A code completion dropdown menu is open, listing several HTML element types:

- Home const Home: () => JSX.Element
- HTMLFormElement
- HTMLModElement
- HTMLAudioElement
- HTMLVideoElement
- HTMLVideoElementEventMap
- HTMLDocument
- HTMLBodyElement
- HTMLBodyElementEventMap
- HTMLFontElement
- HTMLSlotElement

A screenshot of the Visual Studio Code interface. The title bar shows "About.js - react-redux-complete - Visual Studio Code". The left sidebar is the Explorer view, displaying the project structure:

- OPEN EDITORS: App.js, Home.js, About.js
- REACT-REDUX-COMPLETE: poketimes (with node_modules, public, src), components (with About.js, Home.js, App.js, index.css, index.js, registerServiceWorker.js), .gitignore, package-lock.json, package.json, README.md.

The main editor area contains the following code for the About.js component:

```
1 import React from 'react'
2
3 const About = () => {
4     return (
5         <div className="container">
6             <h4 className="center">About</h4>
7             <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisic
8             </div>
9     )
10 }
11
12 export default About;
```

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Contact.js - React-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under "REACT-REDUX-COMPLETE".
 - OPEN EDITORS: App.js, Home.js, About.js, Contact.js
 - REACT-REDUX-COMPLETE:
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md
- Editor View (Right):** The "Contact.js" tab is active, displaying the following code:

```
1 import React from 'react'
2
3 const Contact = () => {
4   return (
5     <div className="container">
6       <h4 className="center">Contact</h4>
7       <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci
8       </div>
9   )
10 }
11
12 export default Contact
```

Navbar.js - react-reduce-complete - Visual Studio Code

EXPLORER

- OPEN EDITORS
- App.js poketimes/src
- Home.js poketimes/src/components
- About.js poketimes/src/components
- Contact.js poketimes/src/components
- Navbar.js poketimes/src/components

REACT-REDUX-COMPLETE

- poketimes
- node_modules
- public
- src
- components
- About.js
- Contact.js
- Home.js
- Navbar.js

- App.js
- index.css
- index.js
- registerServiceWorker.js
- .gitignore
- package-lock.json
- package.json
- README.md

```
1 import React from 'react'
2
3 const Navbar = () => {
4   return (
5     <nav className="nav-wrapper red darken-3">
6       <div className="container">
7         <a className="brand-logo">Poke'Times</a>
8         <ul className="right">
9           <li><a href="/">Home</a></li>
10          <li><a href="/about">About</a></li>
11          <li><a href="/contact">Contact</a></li>
12        </ul>
13      </div>
14    </nav>
15  )
16}
17
18 export default Navbar
```

(Navbar const Navbar: () => JSX...)

```
App.js - react-redux-complete - Visual Studio Code
```

EXPLORER

- OPEN EDITORS 1 UNSAVED
- App.js poketimes/src
- Home.js poketimes/src/components
- About.js poketimes/src/components
- Contact.js poketimes/src/components
- Navbar.js poketimes/src/components

REACT-REDUX-COMPLETE

- poketimes
- node_modules
- public
- src
- components
- About.js
- Contact.js
- Home.js
- Navbar.js
- App.js
- index.css
- Index.js
- registerServiceWorker.js

.gitignore

package-lock.json

package.json

README.md

```
App.js
```

1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3
4 class App extends Component {
5 render() {
6 return (
7 <div className="App">
8
9 </div>
10);
11 }
12 }
13
14 export default App;

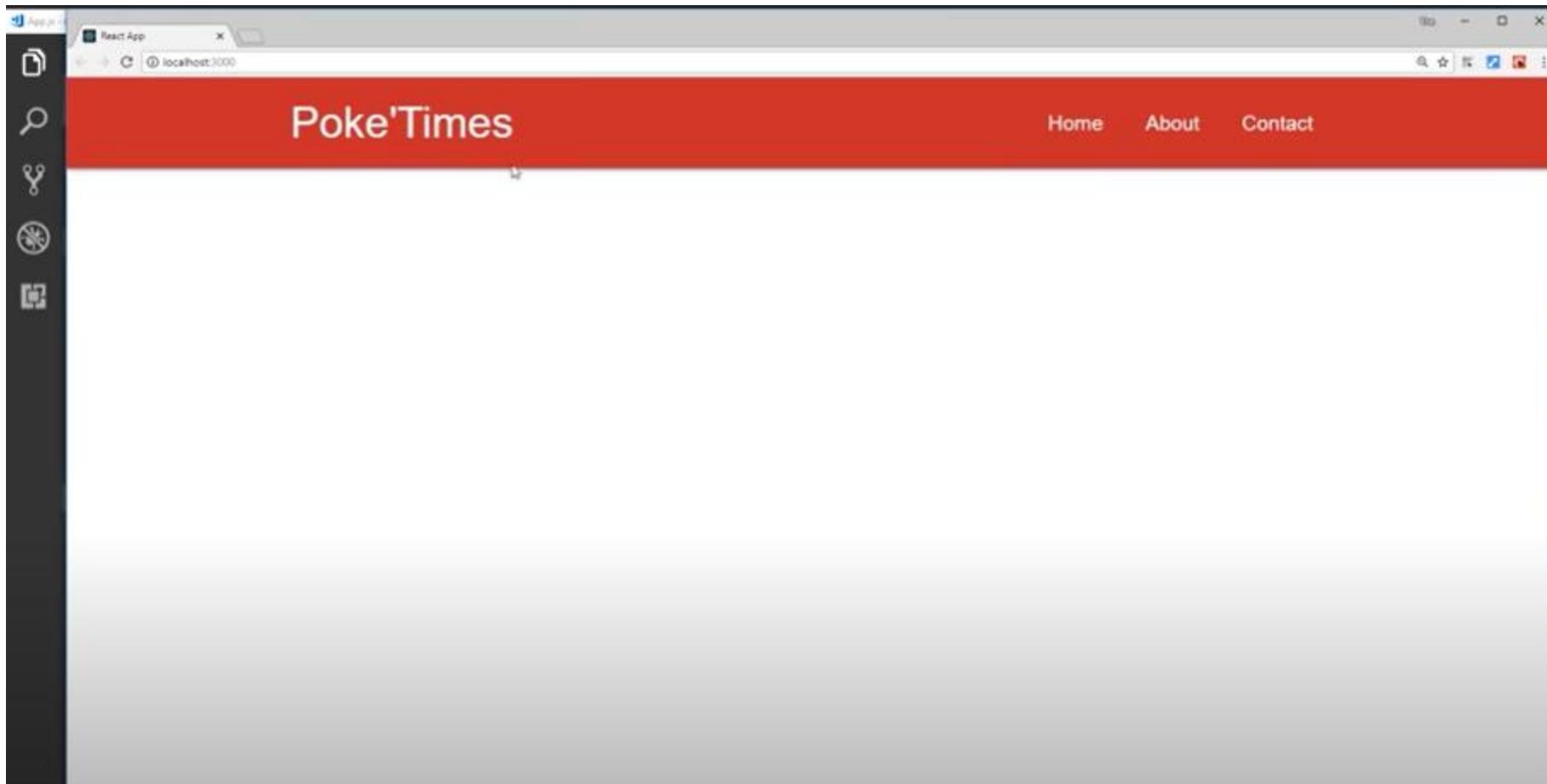
Navbar

Navbar

The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - React-redux-completa - Visual Studio Code
- Explorer Sidebar:** Shows the project structure under "REACT-REDUX-COMPLETE".
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md
- Editor Area:** Displays the code for App.js.

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3
4 class App extends Component {
5   render() {
6     return (
7       <div className="App">
8         <Navbar />
9       </div>
10    );
11  }
12}
13
14 export default App;
15
```



The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under "REACT-REDUX-COMPLETE". The "src" folder contains "components" which includes "About.js", "Contact.js", "Home.js", and "Navbar.js". The "App.js" file is currently selected.
- Editor View (Right):** Displays the content of the "App.js" file. The code uses ES6 syntax and imports components from "react" and "react-router-dom". It defines a class-based component "App" that renders a "BrowserRouter" component. Inside the "BrowserRouter", there is a "div" with the class name "App" containing a "Navbar" component and a "Route" component for the path "/" pointing to the "Home" component.

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'

5
6 class App extends Component {
7   render() {
8     return (
9       <BrowserRouter>
10      <div className="App">
11        <Navbar />
12        <Route path='/' component={Home} />
13      </div>
14    );
15  }
16}
17
18
19 export default App;
20
```



EXPLORER

- OPEN EDITORS
 - App.js poketimes/src
 - Home.js poketimes/src/components
 - About.js poketimes/src/components
 - Contact.js poketimes/src/components
 - Navbar.js poketimes/src/components
- REACT-REDUX-COMPLETE
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md

App.js

Home.js

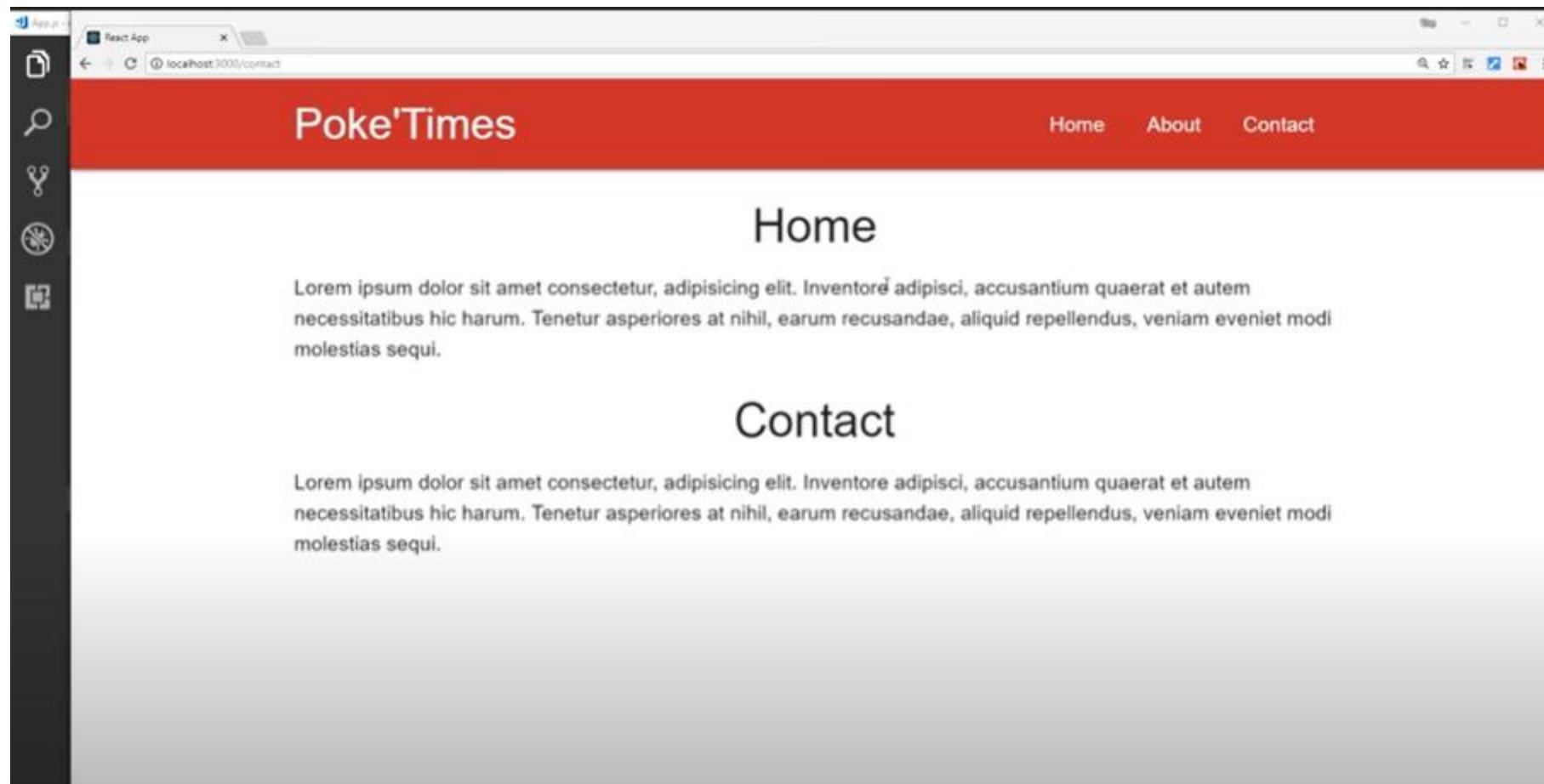
About.js

Contact.js

Navbar.js

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'

7
8 class App extends Component {
9   render() {
10     return (
11       <BrowserRouter>
12         <div className="App">
13           <Navbar />
14           <Route path='/' component={Home} />
15           <Route path='/about' component={About} />
16           <Route path='/contact' component={Contact} />
17         </div>
18       </BrowserRouter>
19     );
20   }
21 }
22
23 export default App;
```



Poke'Times

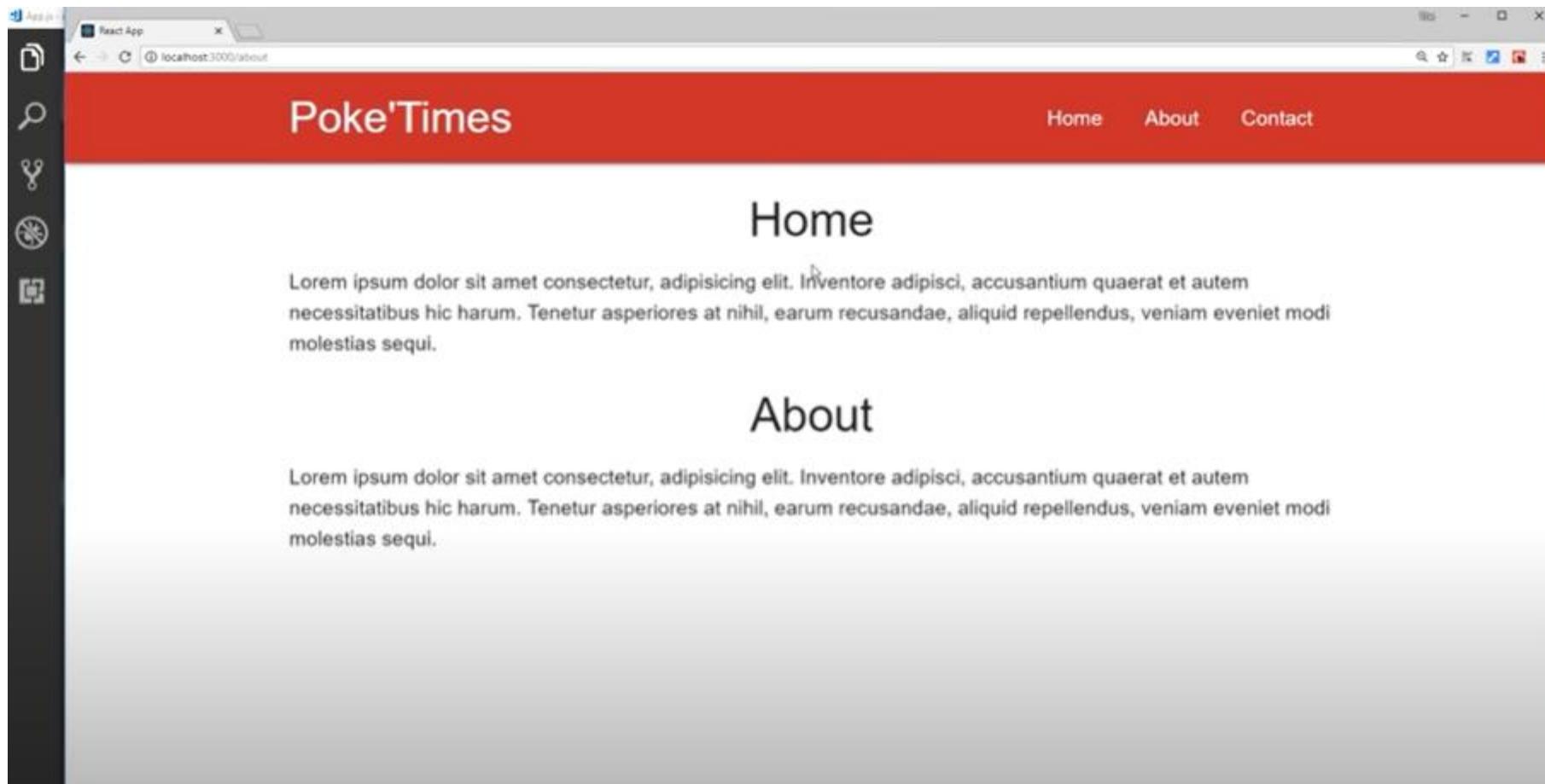
[Home](#) [About](#) [Contact](#)

Home

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil, earum recusandae, aliquid repellendus, veniam eveniet modi molestias sequi.

Contact

Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil, earum recusandae, aliquid repellendus, veniam eveniet modi molestias sequi.



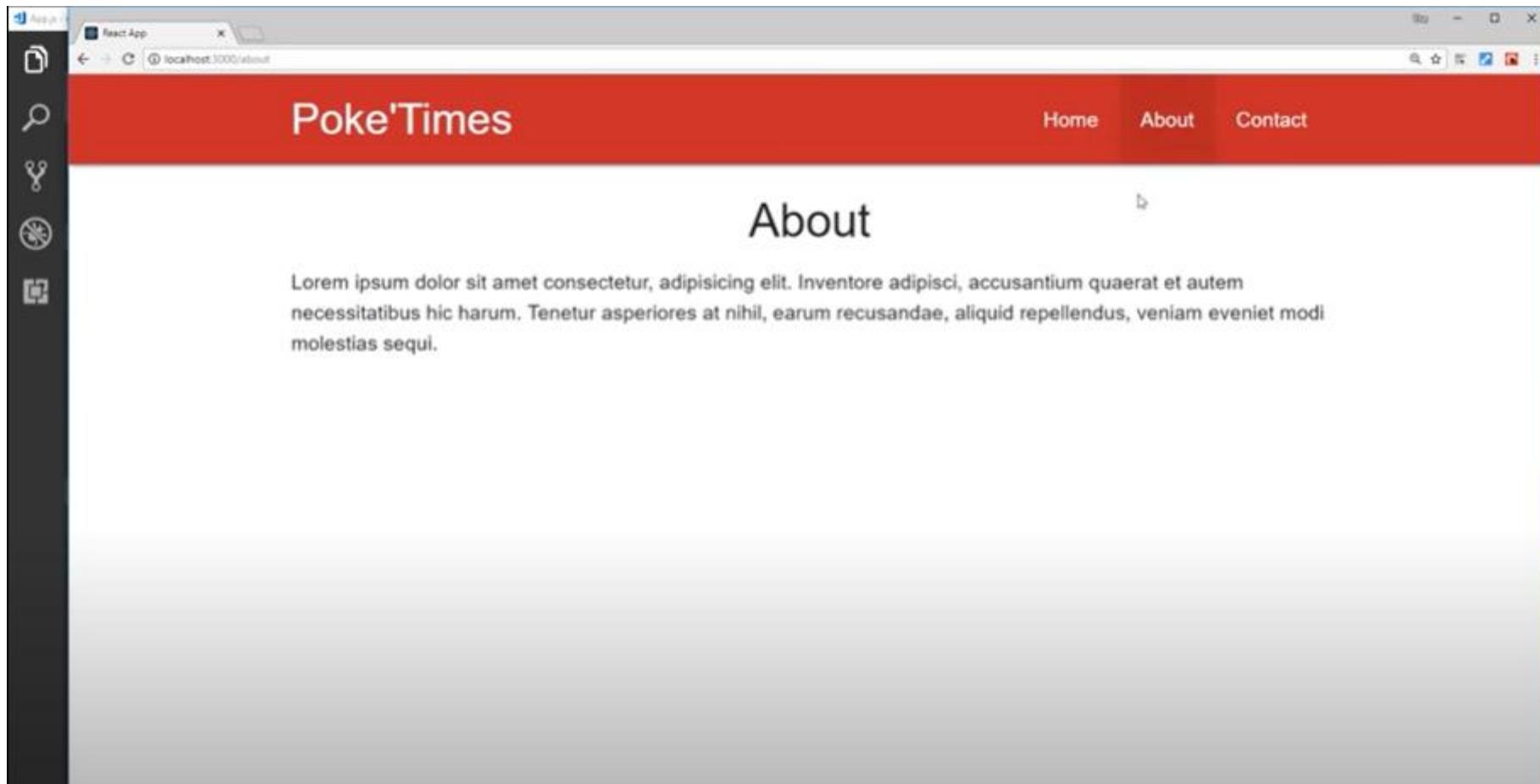
The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under "REACT-REDUX-COMPLETE".
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md
- Open Editors (Top):** App.js, Home.js, About.js, Contact.js, Navbar.js
- Code Editor (Right):** The content of the App.js file is displayed.

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'

7
8 class App extends Component {
9   render() {
10     return (
11       <BrowserRouter>
12         <div className="App">
13           <Navbar />
14           <Route path='/' component={Home} />
15           <Route path='/about' component={About} />
16           <Route path='/contact' component={Contact} />
17         </div>
18       </BrowserRouter>
19     );
20   }
21 }
22
23 export default App;
```

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'
7
8 class App extends Component {
9   render() {
10     return (
11       <BrowserRouter>
12         <div className="App">
13           <Navbar />
14           <Route path='/home' component={Home} />
15           <Route path='/about' component={About} />
16           <Route path='/contact' component={Contact} />
17         </div>
18       </BrowserRouter>
19     );
20   }
21 }
```



The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** Navbar.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under "REACT-REDUX-COMPLETE". The "src" folder contains "components" which includes "About.js", "Contact.js", "Home.js", and "Navbar.js". Other files in "src" include "App.js", "index.css", "index.js", and "registerServiceWorker.js". Global files shown are ".gitignore", "package-lock.json", "package.json", and "README.md".
- Editor View (Right):** Displays the code for "Navbar.js". The code defines a functional component "Navbar" that returns a navigation bar with a logo and three links: Home, About, and Contact.

```
1 import React from 'react'
2
3 const Navbar = () => {
4   return (
5     <nav className="nav-wrapper red darken-3">
6       <div className="container">
7         <a className="brand-logo">Poke'Times</a>
8         <ul className="right">
9           <li><a href="/home">Home</a></li>
10          <li><a href="/about">About</a></li>
11          <li><a href="/contact">Contact</a></li>
12        </ul>
13      </div>
14    </nav>
15  )
16 }
17
18 export default Navbar
```

The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure:
 - OPEN EDITORS: App.js (UNSAVED)
 - REACT-REDUX-COMPLETE: poketimes
 - src: components (About.js, Contact.js, Home.js, Navbar.js), App.js, index.css, index.js, registerServiceWorker.js, .gitignore, package-lock.json, package.json, README.md
- Editor View (Right):** The content of the App.js file is displayed, which defines a class-based component named App that uses the BrowserRouter and Route components from react-router-dom to render different pages based on the URL path.

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'

7
8 class App extends Component {
9   render() {
10     return (
11       <BrowserRouter>
12         <div className="App">
13           <Navbar />
14           <Route exact path='/' component={Home} />
15           <Route path='/about' component={About} />
16           <Route path='/contact' component={Contact} />
17         </div>
18       </BrowserRouter>
19     );
20   }
21 }
22
23 export default App;
```

LINKS & NAVLINKS

```
1 import React from 'react'
2 import { Link, NavLink } from 'react-router-dom'
3
4 const Navbar = () => {
5   return (
6     <nav className="nav-wrapper red darken-3">
7       <div className="container">
8         <a className="brand-logo">Poke'Times</a>
9         <ul className="right">
10           <li><Link to="/">Home</Link></li>
11           <li><Link to="/about">About</Link></li>
12           <li><a href="/contact">Contact</a></li>
13         </ul>
14       </div>
15     </nav>
16   )
17 }
18
19 export default Navbar
```

```
1 import React from 'react'
2 import { Link, NavLink } from 'react-router-dom'
3
4 const Navbar = () => {
5   return (
6     <nav className="nav-wrapper red darken-3">
7       <div className="container">
8         <a className="brand-logo">Poke'Times</a>
9         <ul className="right">
10           <li><Link to="/">Home</Link></li>
11           <li><NavLink to="/about">About</NavLink></li>
12           <li><NavLink to="/contact">Contact</NavLink></li>
13         </ul>
14       </div>
15     </nav>
16   )
17 }
18
19 export default Navbar
```

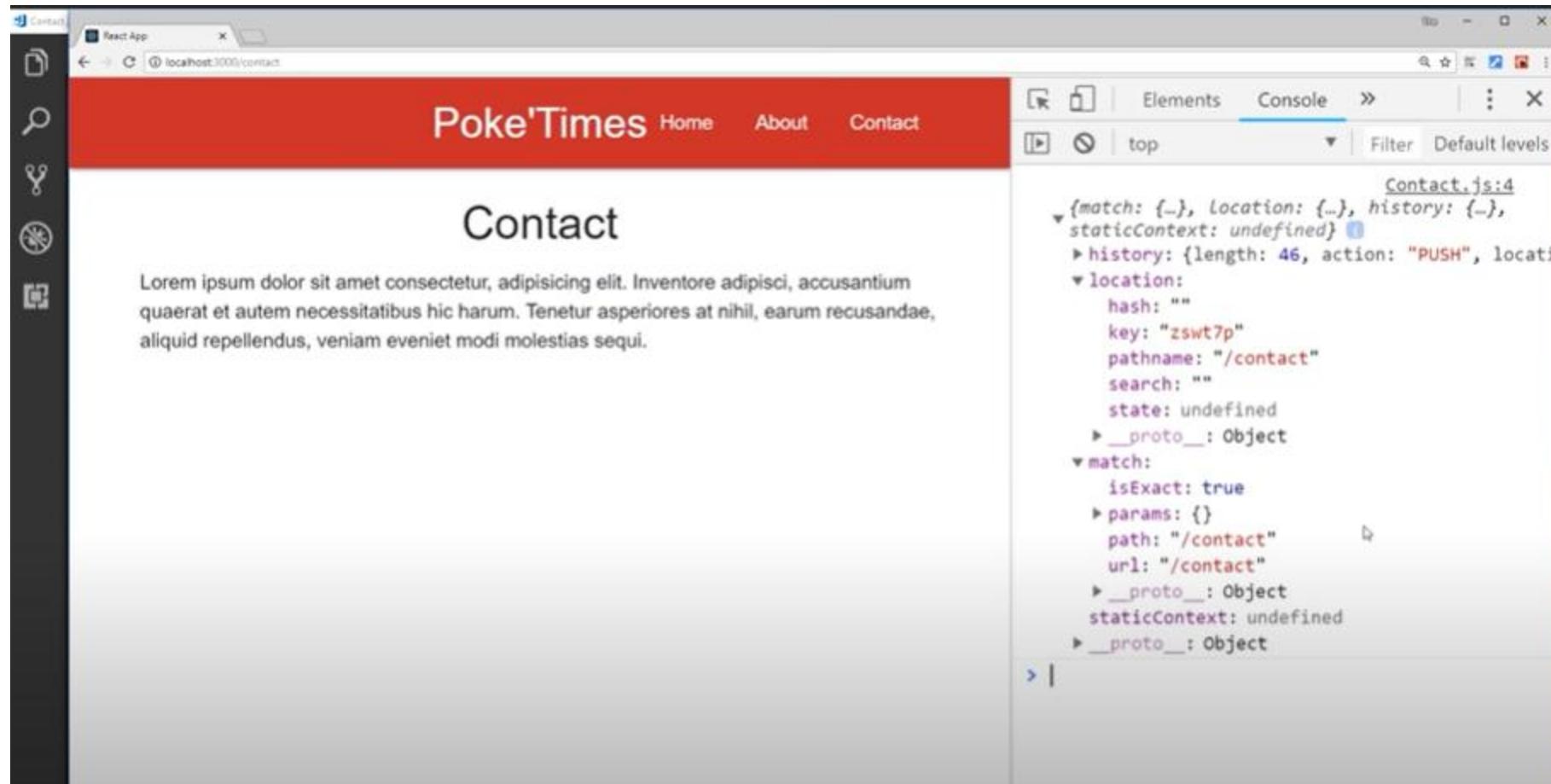
The screenshot shows a browser window displaying the 'Contact' page of a React application named 'Poke'Times'. The page has a red header with the title 'Poke'Times' and navigation links for 'Home', 'About', and 'Contact'. The main content area contains a placeholder text paragraph. On the right side, the browser's developer tools are open, specifically the 'Elements' tab. The DOM tree shows the structure of the page, including the 'root' div, 'App' component, 'nav' bar, and the 'right' ul menu. The 'Contact' link in the menu is highlighted with a blue background and is identified as the active element. The 'Styles' panel on the right lists the CSS rules applied to the selected element, which includes styles from 'materialize.css' and 'element.style.css', such as 'color: #fff;', 'display: block;', and 'padding: 0;'. A note in the top right of the developer tools indicates that JavaScript needs to be enabled to run the app.

```
You need to enable JavaScript to run this app.  
</noscript>  
▼<div id="root">  
  ▼<div class="App">  
    ▼<nav class="nav-wrapper red darken-3">  
      ▼<div class="container">  
        <a class="brand-logo">Poke'Times</a>  
        ▼<ul class="right">  
          ▼<li>  
            <a href="/">Home</a>  
          </li>  
          ▼<li>  
            <a href="/about">About</a>  
          </li>  
          ▼<li>  
            <a href="/contact" class="active" aria-current="page">Contact</a>  
          </li>  
        </ul>
```

:hover .cls {
 element.style {
 color: #fff;
 display: block;
 padding: 0;
 }
}
materialize...
nav ul a {
 -webkit-
 transition:
 background-color: #fff;
 color: #fff;
 font-size: 1rem;
 padding: 0;
 transition:
 background-color: #fff;
 color: #fff;
 font-size: 1rem;
 padding: 0;
}

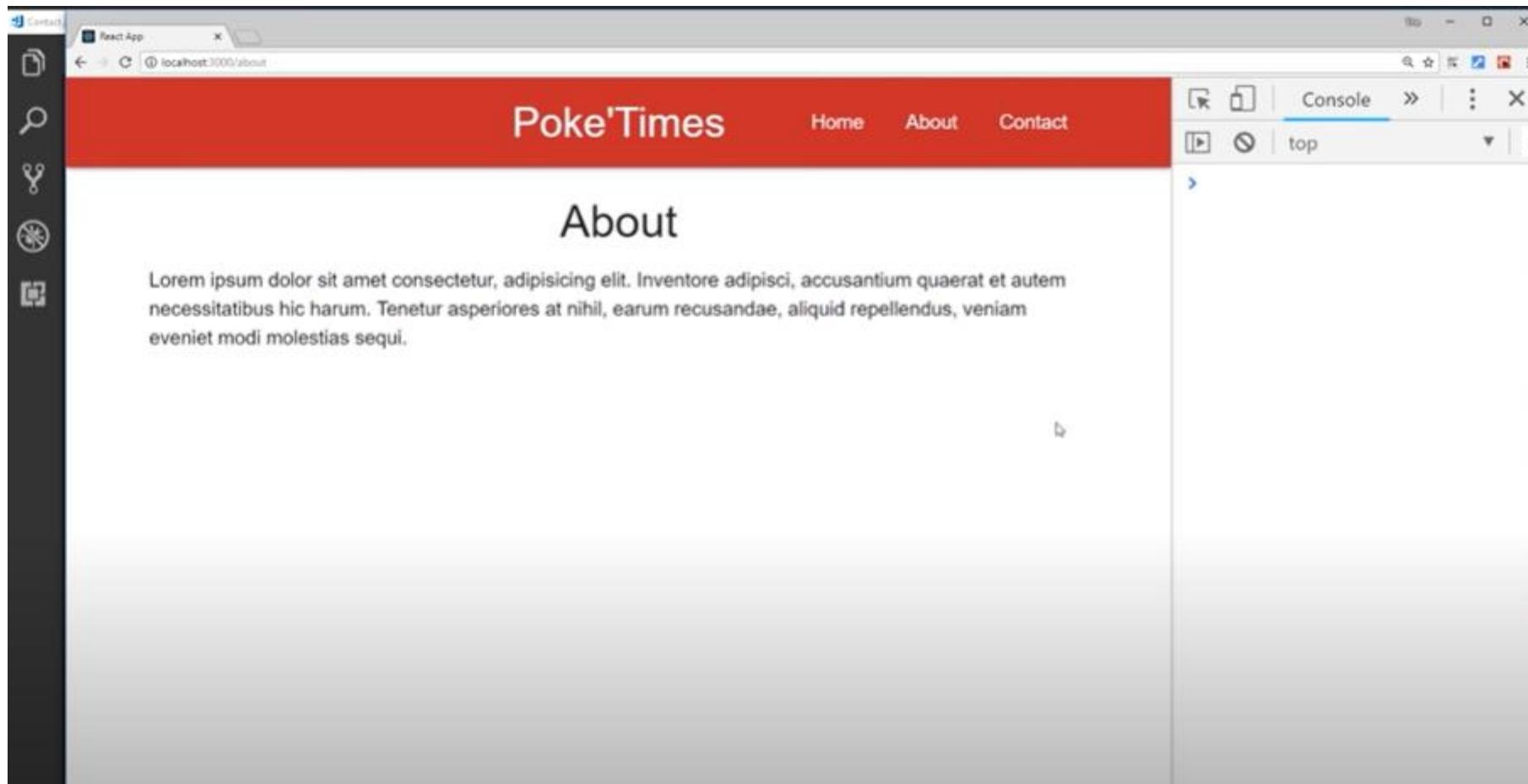
PROGRAMMATIC REDIRECTS

```
1 import React from 'react';
2
3 const Contact = (props: { message?: string; }) => {
4   console.log(props);
5   return (
6     <div className={props.className}>
7       <h4>Payment Response</h4>
8       <p>Lorem ipsum...</p>
9     </div>
10  )
11}
12
13 export default Contact
```



A screenshot of the Visual Studio Code interface, showing the code editor with a dark theme. The title bar reads "Contact.js - React-redux-complete - Visual Studio Code". The tabs at the top show "App.js", "Home.js", "About.js", "Contact.js", and "Navbar.js". The code editor displays the following JavaScript code:

```
1 import React from 'react'
2
3 const Contact = (props) => {
4   setTimeout(() => {
5     props.history.push('/about')
6   }, 2000);
7   return (
8     <div className="container">
9       <h4 className="center">Contact</h4>
10      <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium quaera
11    </div>
12  )
13}
14
15 export default Contact
```

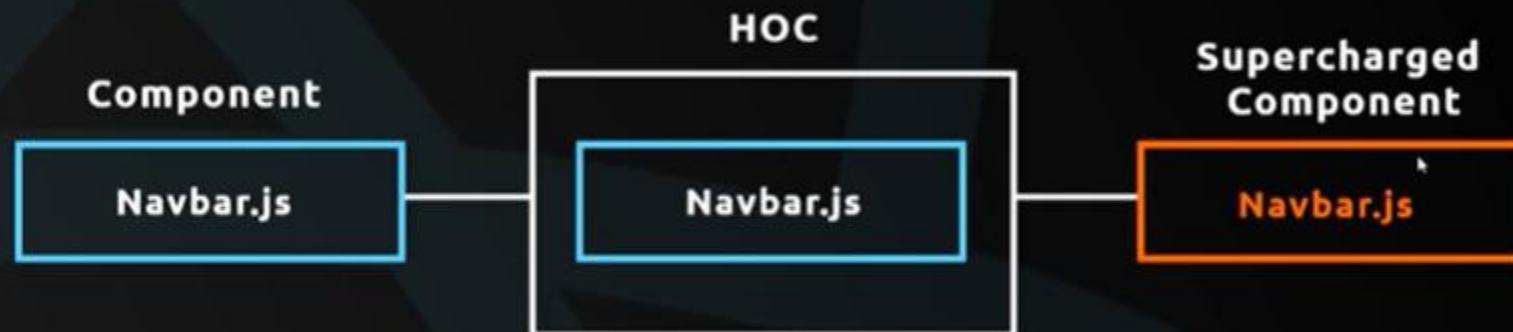


A screenshot of the Visual Studio Code interface showing the `Navbar.js` file. The code implements a navigation bar with a delay before pushing the '/about' route.

```
1 import React from 'react'
2 import { Link, NavLink, withRouter } from 'react-router-dom'
3
4 const Navbar = (props) => {
5   setTimeout(() => {
6     props.history.push('/about')
7   }, 2000)
8   return (
9     <nav className="nav-wrapper red darken-3">
10      <div className="container">
11        <a className="brand-logo">Poke'Times</a>
12        <ul className="right">
13          <li><Link to="/">Home</Link></li>
14          <li><NavLink to="/about">About</NavLink></li>
15          <li><NavLink to="/contact">Contact</NavLink></li>
16        </ul>
17      </div>
18    </nav>
19  )
20}
21
22 export default withRouter(Navbar)
```

HIGHER ORDER COMPONENTS

Higher Order Components



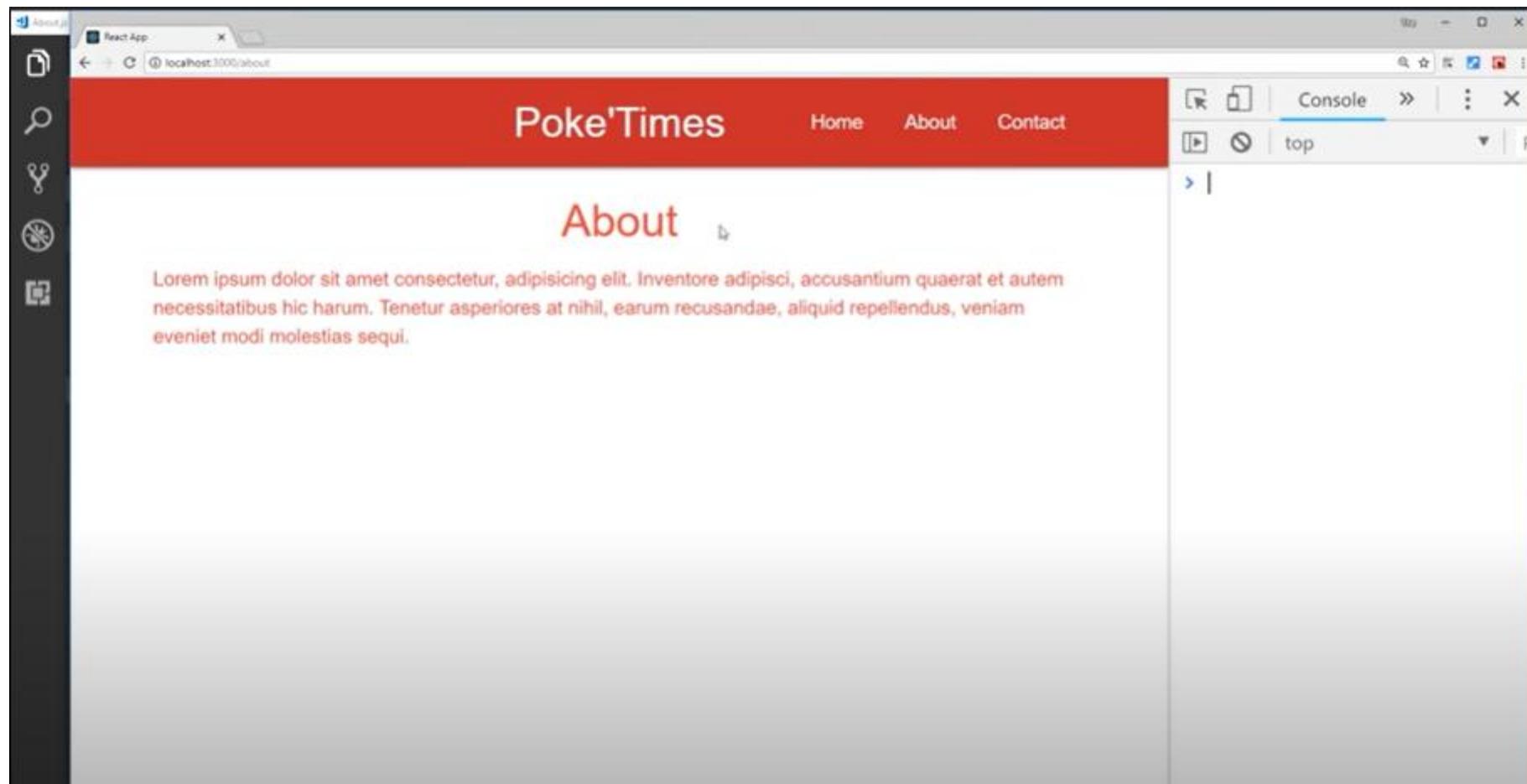
```
1 import React from 'react'
2
3 const Rainbow = (WrappedComponent) => {
4
5     const colours = ['red', 'pink', 'orange', 'blue', 'green', 'yellow'];
6     const randomColour = colours[Math.floor(Math.random() * 5)];
7     const className = randomColour + '-text';
8
9     return (props) => {
10         return (
11             <div className={className}>
12                 <WrappedComponent {...props} />
13             </div>
14         )
15     }
16
17 }
18
19 export default Rainbow
```

Rainbow.js

A screenshot of the Visual Studio Code interface showing the code editor and file explorer. The code editor displays the file `About.js` with the following content:

```
1 import React from 'react'
2 import Rainbow from '../hoc/Rainbow'
3
4 const About = () => {
5   return (
6     <div className="container">
7       <h4 className="center">About</h4>
8       <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci
9       </div>
10  )
11}
12
13 export default Rainbow(About)
```

The file explorer on the left shows the project structure under the `REACT-REDUX-COMPLETE` folder, including files like `App.js`, `Home.js`, `Rainbow.js`, `About.js`, and `Navbar.js`. The `src` folder contains `components`, `hoc`, and `index.css`.



USING AXIOS

To get some data from REST APIs, your server, Firebase (direct database)

Lets get data from a testing REST API :

<https://jsonplaceholder.typicode.com/posts>

The screenshot shows a Visual Studio Code interface with a dark theme. In the center-left pane, there is a code editor with a file named "Home.js". The code is as follows:

```
1 import React from 'react'
2
3 const Home = () => {
4   return (
5     <div className="container">
6       <h4 className="center">Home</h4>
7       <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium
8         quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil, earum recusandae,
9         aliquid repellendus, veniam eveniet modi molestias sequi.</p>
10    </div>
11  )
12
13 export default Home
```

Below the code editor, there is a navigation bar with tabs: PROBLEMS, OUTPUT, DEBUG CONSOLE, and TERMINAL. The TERMINAL tab is selected, showing a Windows PowerShell session. The output of the session is:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\Shaun\Documents\Tutorials\Recording\react-redux-complete> cd poketimes
PS C:\Users\Shaun\Documents\Tutorials\Recording\react-redux-complete\poketimes> npm install axios
npm WARN ajv-keywords@3.2.0 requires a peer of ajv@^6.0.0 but none is installed. You must install peer dependencies yourself.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.4 (node_modules\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.4: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
```

Component lifehooks only used in class based component

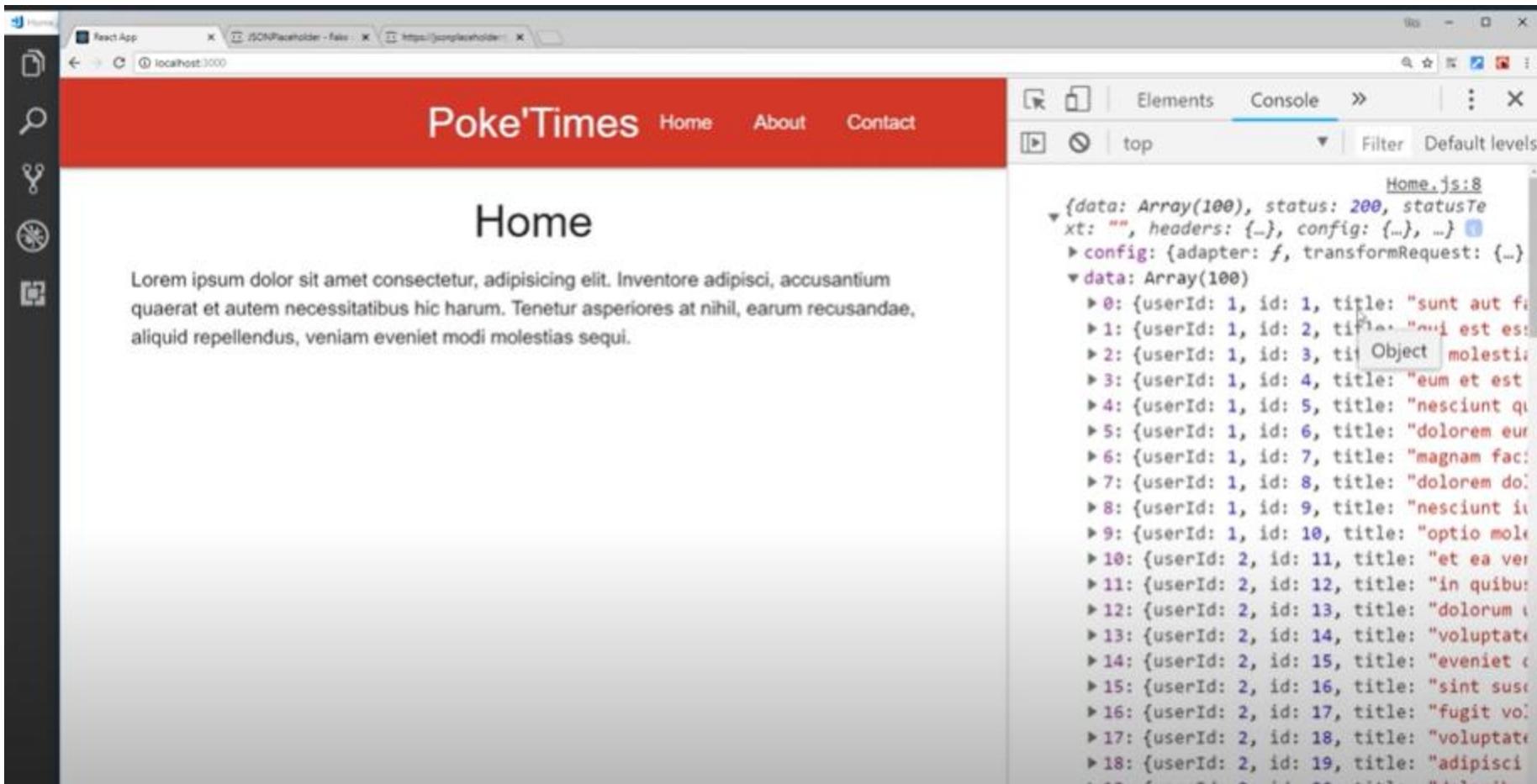
```
ComponentDidMount(){  
}
```

component mounts using the useEffect hook in function based component.

```
useEffect(() => {  
}, []);
```

A screenshot of the Visual Studio Code interface showing a file named "Home.js". The code is a class component that imports React and axios. It uses axios to get data from a JSONPlaceholder endpoint and logs the response to the console. The render method returns a container div containing a centered h4 and a paragraph of lorem ipsum text.

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3
4 class Home extends Component {
5   componentDidMount(){
6     axios.get('https://jsonplaceholder.typicode.com/posts')
7       .then(res => {
8         console.log(res)
9       })
10    }
11  }
12  render() {
13    return (
14      <div className="container">
15        <h4 className="center">Home</h4>
16        <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium
17          quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil, earum recusandae,
18          aliquid repellendus, veniam eveniet modi molestias sequi.</p>
19      </div>
20    )
21  }
}
```



SHOWING DATA IN SOME TEMPLATE

Lets have posts empty list

And add the res in that list

A screenshot of the Visual Studio Code interface showing a file named 'Home.js'. The code is a React component that fetches data from an API and renders it. A tooltip or code completion dropdown is visible over the line 'posts: res.data.slice(0,10)'. The code is as follows:

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3
4 class Home extends Component {
5   state = {
6     posts: [ ]
7   }
8   componentDidMount(){
9     axios.get('https://jsonplaceholder.typicode.com/posts')
10    .then(res => {
11      console.log(res)
12      this.setState({
13        posts: res.data.slice(0,10)
14      })
15    })
16  }
17 }
18 render() {
19   return (
20     <div className="container">
21       <h4 className="center">Home</h4>
22       <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium  
quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil earum recusandae.
```

A screenshot of the Visual Studio Code interface showing a file named "Home.js". The code is a class-based component named "Home" that extends "Component". It uses the axios library to fetch data from a JSONPlaceholder API endpoint. The component's state contains an array of posts, which it initializes to an empty array. In the componentDidMount() lifecycle method, it makes a GET request to the specified URL. If successful, it logs the response to the console and updates its state with the first 10 posts from the response data. The render() method returns a container div containing a centered h4 element with the text "Home" and a p element with placeholder Latin text (Lorem ipsum). The code is numbered from 1 to 23.

```
3
4  class Home extends Component {
5    state = {
6      posts: [ ]
7    }
8    componentDidMount(){
9      axios.get('https://jsonplaceholder.typicode.com/posts')
10     .then(res => {
11       console.log(res)
12       this.setState({
13         posts: res.data.slice(0,10)
14       })
15     })
16   }
17   render() {
18     const { posts } = this.state;
19     const postList = posts.length ? () : []
20     return (
21       <div className="container">
22         <h4 className="center">Home</h4>
23         <p>Lorem ipsum dolor sit amet consectetur, adipisicing elit. Inventore adipisci, accusantium  
quaerat et autem necessitatibus hic harum. Tenetur asperiores at nihil, earum recusandae,  
aliquid consequatur veniam exceptio modi molestiae sequi.</p>
```

A screenshot of the Visual Studio Code interface showing a file named `Home.js`. The code is a class-based component that fetches data from an API and renders it. The component has a state with an array of posts, which it initializes to an empty array. It then performs a network request using the `axios` library to get data from `'https://jsonplaceholder.typicode.com/posts'`. The response is logged to the console, and the state is updated with the first 10 posts from the JSON response. The `render` method checks if there are posts in the state. If no posts are present, it displays a message in a centered `<div>`. If there are posts, it returns a `<div>` with a `className` of `"container"`, containing a `<h4>` with the text `"Home"` and a paragraph of placeholder text (`"Lorem ipsum dolor sit amet, consectetur adipisicing elit. Inventore adipisci accusantium"`). A tooltip for the variable `pageYOffset` is visible at the bottom of the code editor.

```
3
4  class Home extends Component {
5    state = {
6      posts: [ ]
7    }
8    componentDidMount(){
9      axios.get('https://jsonplaceholder.typicode.com/posts')
10     .then(res => {
11       console.log(res)
12       this.setState({
13         posts: res.data.slice(0,10)
14       })
15     })
16   }
17   render() {
18     const { posts } = this.state;
19     const postList = posts.length ? () : (
20       <div className="center">No posts yet</div>
21     )
22     return (
23       <div className="container">
24         <h4 className="center">Home</h4>
25         <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit. Inventore adipisci accusantium</p>
26       </div>
27     )
28   }
29 }
```

```
Home.js - react-redux-complete - Visual Studio Code
Home.js
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
```

```
    }
    componentDidMount(){
        axios.get('https://jsonplaceholder.typicode.com/posts')
            .then(res => {
                console.log(res)
                this.setState({
                    posts: res.data.slice(0,10)
                })
            })
    }
    render() {
        const { posts } = this.state;
        const postList = posts.length ? (
            posts.map(post => {
                return (
                    <div className="post card" key={post.id}>
                        <div className="card-content">
                            <span className="card-title">{post.title}</span>
                            <p>{post.body}</p>
                        </div>
                    </div>
                )
            })
        )
        return (
            <div>
                <h1>Posts</h1>
                <hr/>
                {postList}
            </div>
        )
    }
}
```

A screenshot of the Visual Studio Code interface showing the `Home.js` file. The code is a functional component that maps over an array of posts to render a list of cards. If there are no posts, it displays a message. The code uses JSX syntax with class names like `post card`, `card-content`, `card-title`, and `center`. A tooltip is visible at the bottom of the screen, pointing to the `postList` variable.

```
const postList = posts.length ? \n  posts.map(post => {\n    return (\n      <div className="post card" key={post.id}>\n        <div className="card-content">\n          <span className="card-title">{post.title}</span>\n          <p>{post.body}</p>\n        </div>\n    </div>\n  )\n})\n: (\n  <div className="center">No posts yet</div>\n)\nreturn (\n  <div className="container">\n    <h4 className="center">Home</h4>\n    {postList}\n  </div>\n)\n}
```

Poke'Times Home About Contact

Home

sunt aut facere repellat provident occaecati
excepturi optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit
molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores
neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam
non debitis possimus qui neque nisi nulla

```
Home.js:11
▶ {data: Array(100), status: 200, statusText: "", headers: {...}, config: {...}, ...}
```

ROUTE PARAMETERS

Route Parameters

- Part of a URL that can change
- Usually see them on URLs for individual records

User Profile Route:

mysite.com/users/[yoshi2k1](#)
mysite.com/users/[mariotheman](#)

Cooking recipe page:

mycookingsite.com/recipes/[12345](#)
mycookingsite.com/recipes/[98765](#)

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under "REACT-REDUX-COMPLETE".
 - OPEN EDITORS: 1 UNSAVED (Home.js)
 - Home.js (poketimes/src/components)
 - App.js (poketimes/src)
 - Post.js (poketimes/src/components)

REACT-REDUX-COMPLETE

 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - gitignore
 - package-lock.json
 - package.json
 - README.md
- Editor View (Right):** The content of the App.js file is displayed.

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'
7 import Post from './components/Post'

8
9 class App extends Component {
10   render() {
11     return (
12       <BrowserRouter>
13         <div className="App">
14           <Navbar />
15           <Route exact path='/' component={Home} />
16           <Route path='/about' component={About} />
17           <Route path='/contact' component={Contact} />
18           <Route path="/:post_id" component={Post} />
19         </div>
20       </BrowserRouter>
21     );
22   }
23 }
```

Post.js - react-movie-complete - Visual Studio Code

EXPLORER

OPEN EDITORS

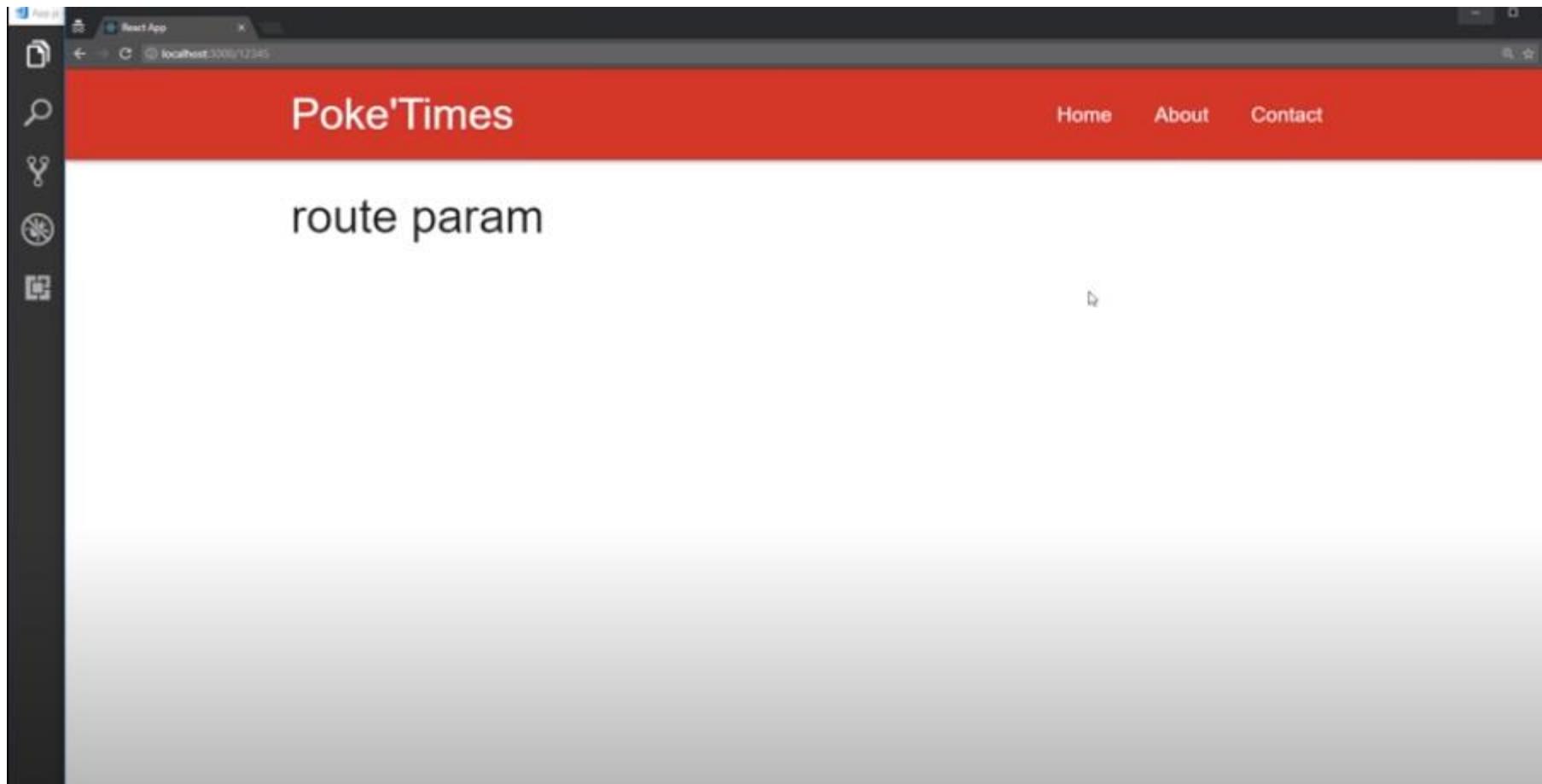
- Home.js poketimes/src/components
- App.js poketimes/src
- Post.js poketimes/src/components

REACT-REDUX-COMPLETE

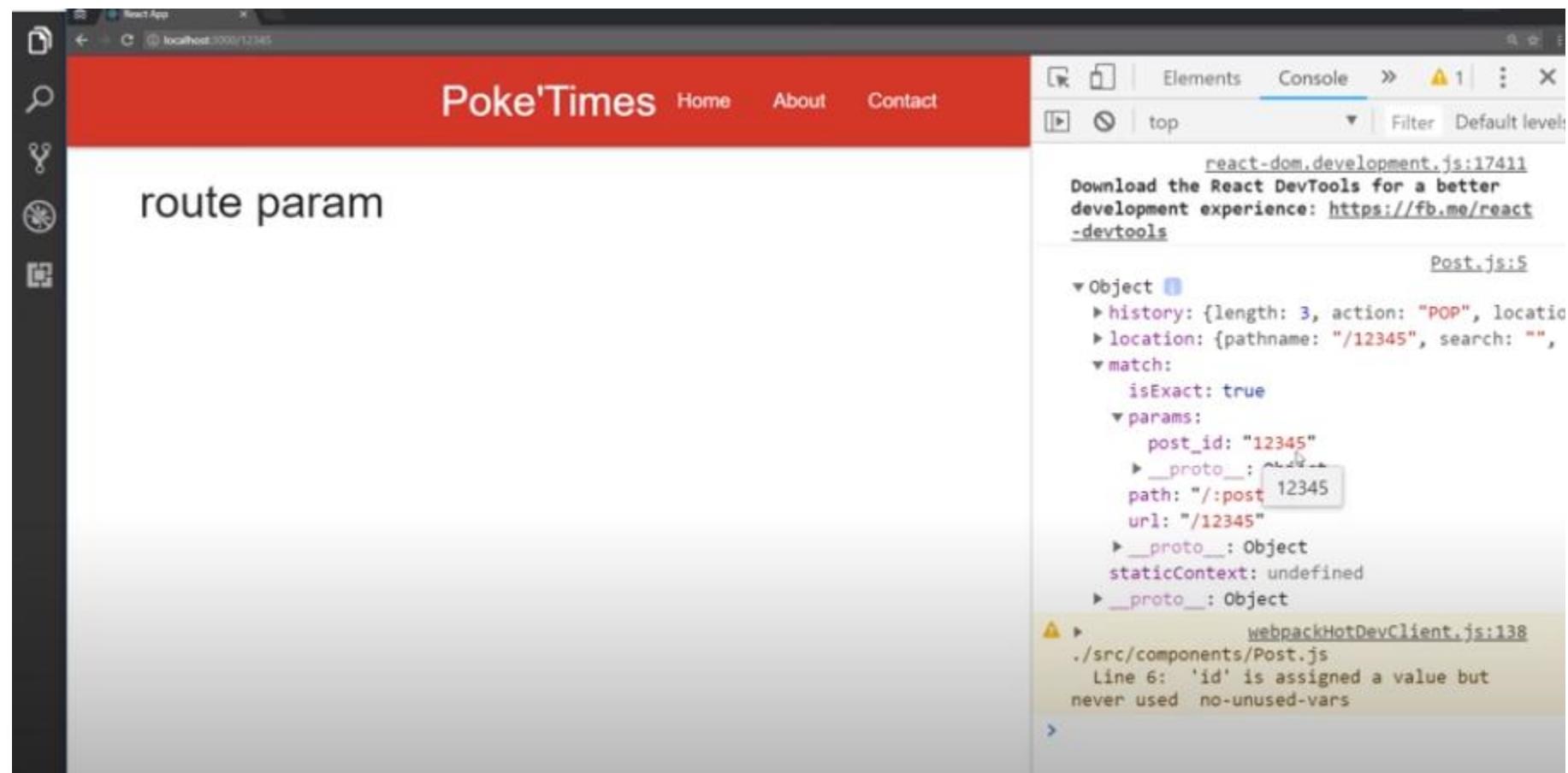
- poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - gitignore
 - package-lock.json
 - package.json
 - README.md

Home.js App.js Post.js

```
1 import React, { Component } from 'react'
2
3 class Post extends Component {
4   render() {
5     return (
6       <div className="container">
7         <h4>route param</h4>
8       </div>
9     )
10  }
11 }
12
13 export default Post
```



```
1 import React, { Component } from 'react'
2
3 class Post extends Component {
4     componentDidMount(){
5         console.log(this.props)
6         let id = this.props.match.params.post_id;
7     }
8     render() {
9         return (
10             <div className="container">
11                 <h4>route param</h4>
12             </div>
13         )
14     }
15 }
16
17 export default Post
```

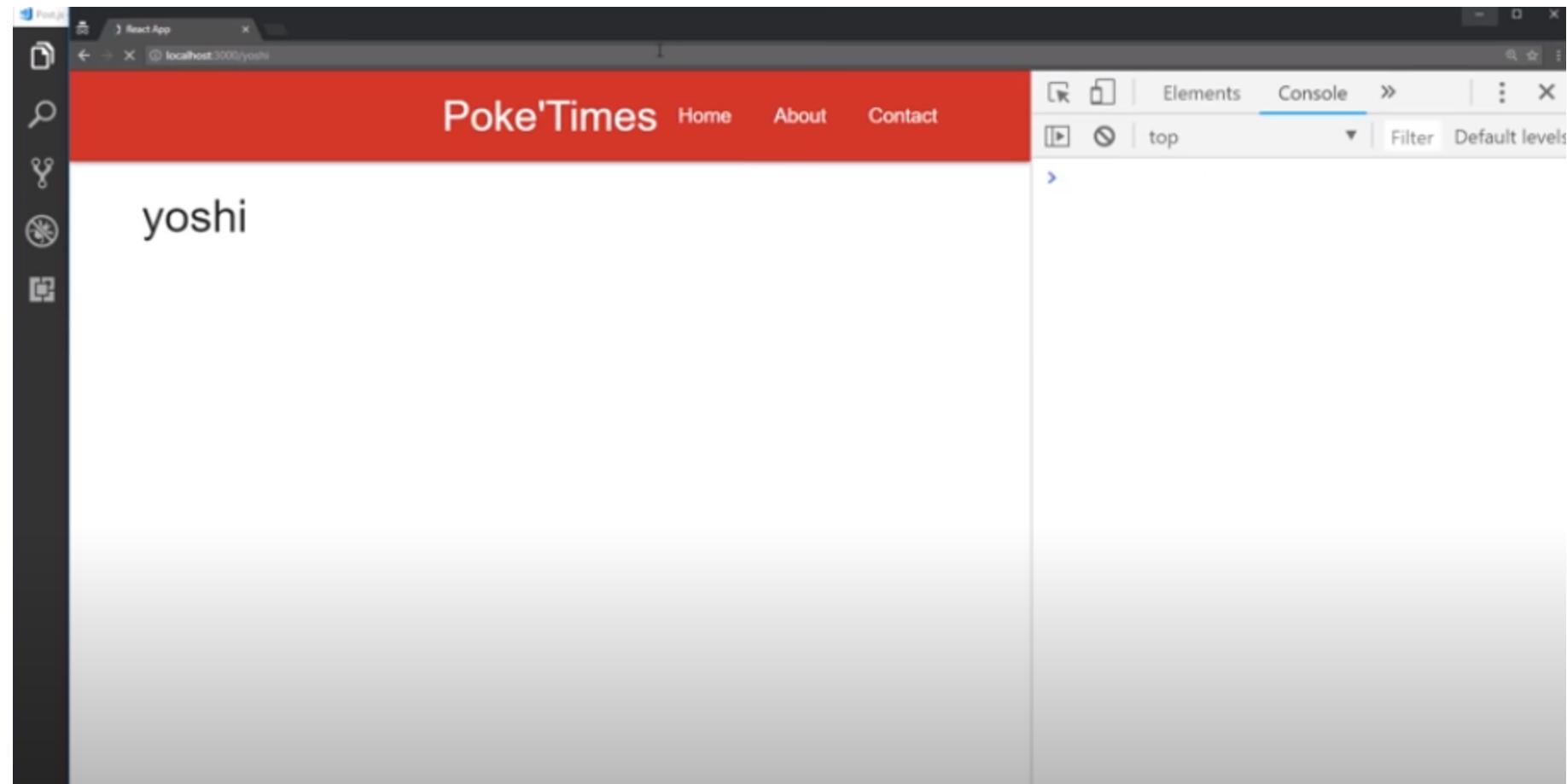


A screenshot of the Visual Studio Code interface, showing the file `Post.js` open in the editor. The code is a React component named `Post` that fetches a post by ID from props and displays it in an `<h4>` tag.

```
1 import React, { Component } from 'react'
2
3 class Post extends Component {
4   state = {
5     id: null
6   }
7   componentDidMount(){
8     let id = this.props.match.params.post_id;
9     this.setState({
10       id: id
11     })
12   }
13   render() {
14     return (
15       <div className="container">
16         <h4>{this.state.id}</h4>
17       </div>
18     )
19   }
20 }
21
22 export default Post
```

The cursor is positioned at the end of the string `{this.state.id}`. A code completion tooltip is displayed, showing the type information for the variable `this.state.id`:

- `id` (property) `id: any`
- `post_id`



The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Home.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure under the 'REACT-REDUX-COMPLETE' workspace:
 - OPEN EDITORS: Home.js (UNSAVED), App.js, Post.js
 - REACT-REDUX-COMPLETE:
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - gitignore
 - package-lock.json
 - package.json
 - README.md
 - Code Editor (Right):** Displays the Home.js file content:

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3 import { Link } from 'react-router-dom'
4
5 class Home extends Component {
6   state = {
7     posts: [ ]
8   }
9   componentDidMount(){
10     axios.get('https://jsonplaceholder.typicode.com/posts')
11       .then(res => {
12         console.log(res)
13         this.setState({
14           posts: res.data.slice(0,10)
15         })
16       })
17   }
18   render() {
19     const { posts } = this.state;
20     const postList = posts.length ? (
21       posts.map(post => {
22         return (
23           <div className="post card" key={post.id}>
```

```
14     posts: res.data.slice(0,10)
15   })
16 }
17 }
18 render() {
19   const { posts } = this.state;
20   const postList = posts.length ? (
21     posts.map(post => {
22       return (
23         <div className="post card" key={post.id}>
24           <div className="card-content">
25             <Link to={'/' + post.id}>
26               <span className="card-title">{post.title}</span>
27             </Link>
28             <p>{post.body}</p>
29           </div>
30         </div>
31       )
32     })
33   ) : (
34     <div className="center">No posts yet</div>
35   )
36   return (

```

The screenshot displays a web browser window with a React application titled "Poke'Times". The application has a red header bar with the title "Poke'Times" and navigation links for "Home", "About", and "Contact". Below the header, there are two cards. The first card contains the text:
**sunt aut facere repellat provident occaecati
excepturi optio reprehenderit**
quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit
molestiae ut ut quas totam nostrum rerum est autem sunt rem eveniet architecto

The second card contains the text:
qui est esse
est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores
neque fugiat blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam
non debitis possimus qui neque nisi nulla

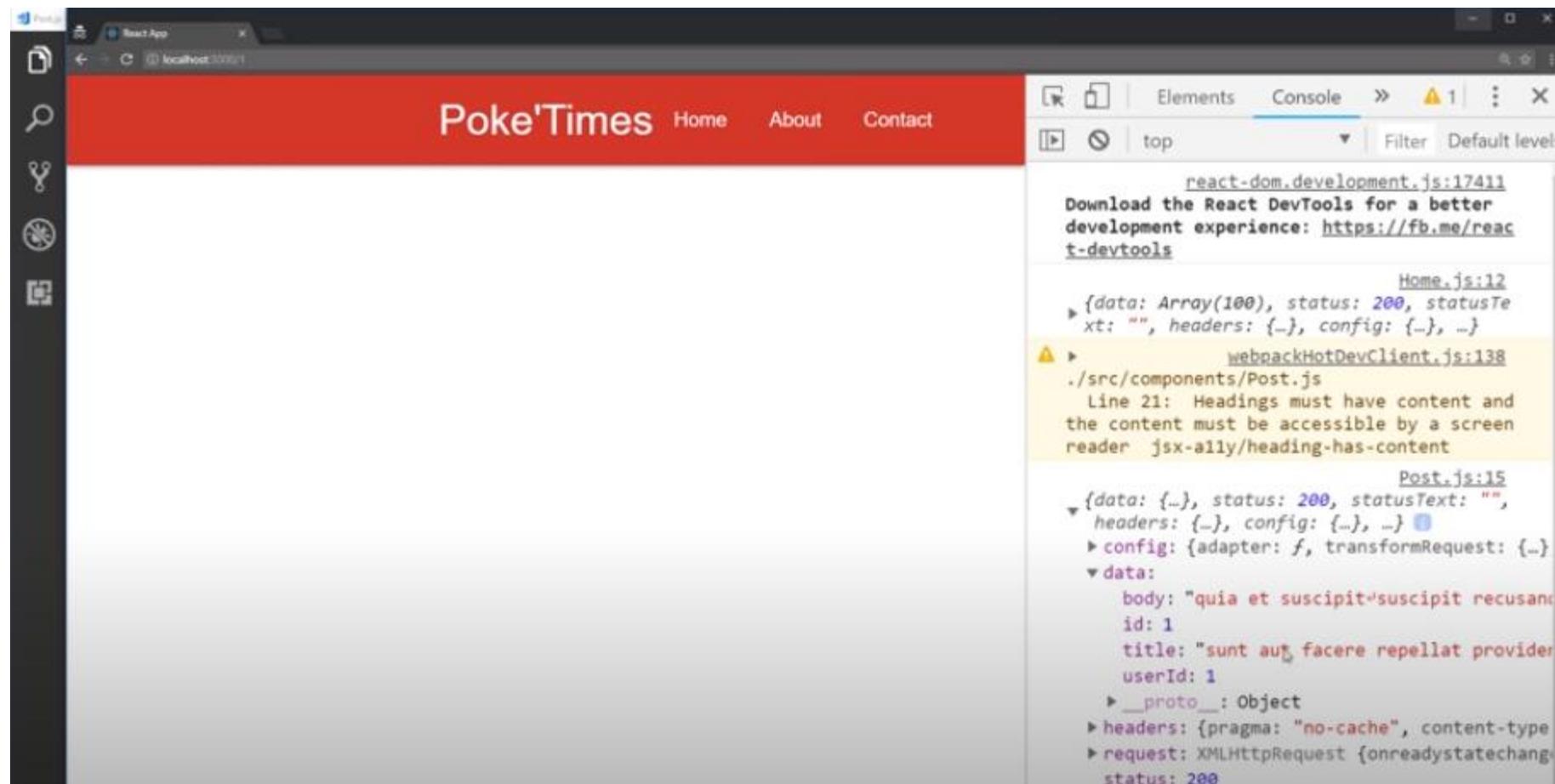
On the right side of the browser window, the developer tools are open, specifically the "Console" tab. It shows a log entry from "react-dom.development.js:17411" with the message: "Download the React DevTools for a better development experience: <https://fb.me/react-devtools>". Below this, it shows a response object from "Home.js:12":

```
{data: Array(100), status: 200, statusText: "", headers: {...}, config: {...}, ...}
```

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Post.js - react-redux-complete - Visual Studio Code
- Explorer View:** Shows the project structure under "REACT-REDUX-COMPLETE". The "src" folder contains "components" (About.js, Contact.js, Home.js, Navbar.js, Post.js), "hoc" (Rainbow.js, App.js, Index.css, index.js, registerServiceWorker.js), and other files (.gitignore, package-lock.json, package.json, README.md).
- Editor View:** Displays the code for "Post.js". The code uses React and axios to fetch a post from a JSONPlaceholder API endpoint.

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3
4 class Post extends Component {
5   state = {
6     post: null
7   }
8   componentDidMount(){
9     let id = this.props.match.params.post_id;
10    axios.get('https://jsonplaceholder.typicode.com/posts/' + id)
11      .then(res => {
12        this.setState({
13          post: res.data
14        })
15        console.log(res)
16      })
17    }
18    render() {
19      return (
20        <div className="container">
21          <h4></h4>
22        </div>
23      )
24    }
25  }
26}
```



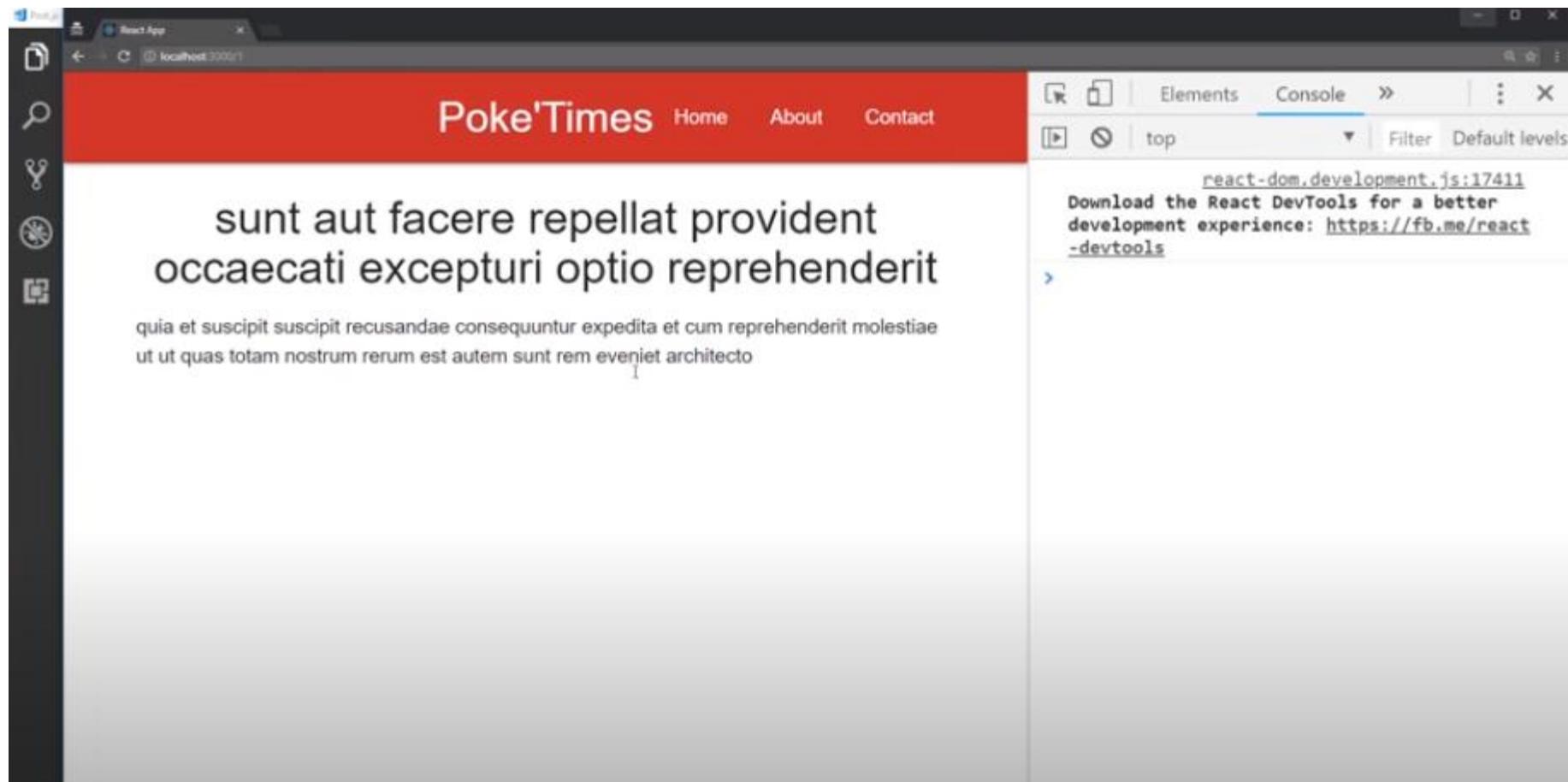
A screenshot of the Visual Studio Code interface. The left sidebar shows the file structure of a React application named 'poketimes'. The current file being edited is 'Post.js', which contains a class-based component. The component has a state variable 'post' and a 'componentDidMount' lifecycle method that uses the 'axios' library to fetch data from 'https://jsonplaceholder.typicode.com'. The 'render' method returns a loading message if 'post' is null. A code completion tooltip is open over the 'post' variable in the 'render' method, listing various suggestions such as 'PerfWidgetExternal', 'PeriodicWave', 'PermissionRequest', 'PermissionRequestedEvent', 'personalbar', 'Plugin', 'PluginArray', 'PointerEvent', 'PopStateEvent', 'Position', 'PositionError', and an ellipsis (...). The code editor has a dark theme.

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3
4 class Post extends Component {
5   state = {
6     post: null
7   }
8   componentDidMount() {
9     let id = this.props.match.params.po
10    axios.get('https://jsonplaceholder.typicode.com/posts/' + id)
11      .then(res => {
12        this.setState({
13          post: res.data
14        })
15      })
16    }
17    render() {
18      const post = this.state.post ? () : null
19      return (
20        <div className="center">Loading post...</div>
21      )
22    }
23  }
24
25  return (
26    <div>
27      <h1>Poketimer</h1>
28      <h2>Post</h2>
29      <Post />
30    </div>
31  )
32}
```

The screenshot shows the Visual Studio Code interface with the following details:

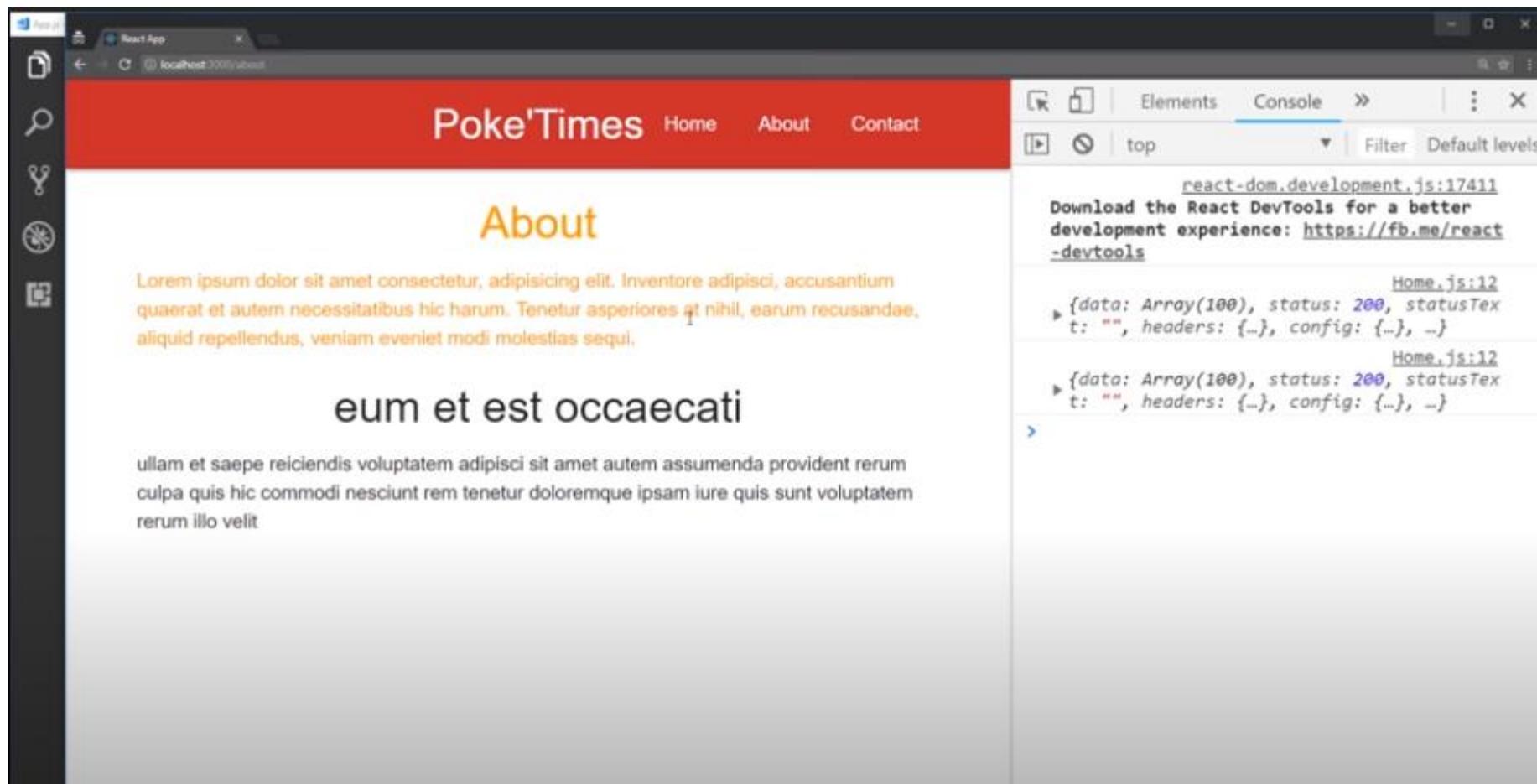
- Title Bar:** Post.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):**
 - OPEN EDITORS: Home.js, App.js, Post.js
 - REACT-REDUX-COMPLETE tree:
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md
- Editor View (Right):** The code for the Post.js component is displayed, showing state management and rendering logic.

```
12     this.setState({
13       post: res.data
14     })
15   })
16 }
17 render() {
18
19   const post = this.state.post ? (
20     <div className="post">
21       <h4 className="center">{this.state.post.title}</h4>
22       <p>{this.state.post.body}</p>
23     </div>
24   ) : (
25     <div className="center">Loading post...</div>
26   )
27
28   return (
29     <div className="container">
30       { post }
31     </div>
32   )
33 }
34 }
```



The screenshot shows a browser window with a React application titled "Poke'Times". The application has a red header bar with the title "Poke'Times" and navigation links for "Home", "About", and "Contact". The main content area is white and displays the word "Contact" in large black font. Below it is a paragraph of placeholder text (Lorem ipsum) and another section with the text "eum et est occaecati". On the right side of the browser window, the developer tools are open, showing the "Console" tab. The console output includes several log messages from "react-dom.development.js" and "Home.js", providing details about component rendering and data status.

```
react-dom.development.js:17411
Download the React DevTools for a better
development experience: https://fb.me/react
-devtools
Home.js:12
▶ {data: Array(100), status: 200, statusText: "", headers: {...}, config: {...}, ...}
Home.js:12
▶ {data: Array(100), status: 200, statusText: "", headers: {...}, config: {...}, ...}
>
```



The screenshot shows a Visual Studio Code window with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View:** Shows the project structure:
 - OPEN EDITORS: Home.js, App.js (selected), Post.js
 - REACT-REDUX-COMPLETE: poketimes (expanded)
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js (selected)
 - index.css
 - index.js
 - registerServiceWorker.js
 - .gitignore
 - package-lock.json
 - package.json
 - README.md- Code Editor:** Displays the content of the App.js file:

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'
7 import Post from './components/Post'

8
9 class App extends Component {
10   render() {
11     return (
12       <BrowserRouter>
13         <div className="App">
14           <Navbar />
15           <Route exact path='/' component={Home} />
16           <Route path='/about' component={About} />
17           <Route path='/contact' component={Contact} />
18           <Route path="/:post_id" component={Post} />
19         </div>
20       </BrowserRouter>
21     );
22   }
23 }
```

SWITCH TAG

The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure:
 - OPEN EDITORS: Home.js, App.js, Post.js
 - REACT-REDUX-COMPLETE folder:
 - poketimes
 - node_modules
 - public
 - src
 - components (Home.js, Contact.js, About.js, Navbar.js, Post.js)
 - hoc (Rainbow.js, App.js, index.css, index.js, registerServiceWorker.js)
 - .gitignore
 - package-lock.json
 - package.json
 - README.md
- Code Editor (Right):** Displays the content of the App.js file:

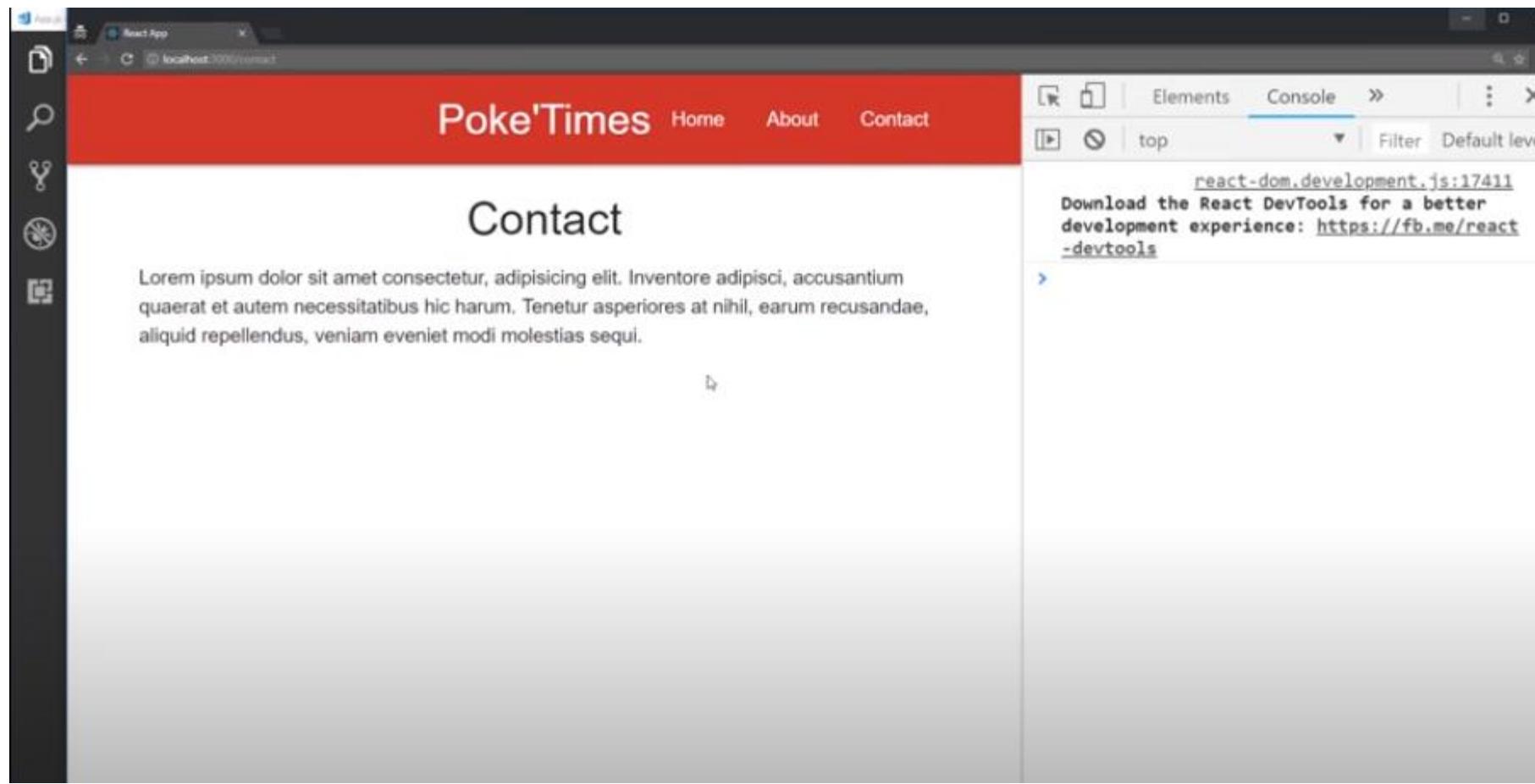
```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'
7 import Post from './components/Post'

8
9 class App extends Component {
10   render() {
11     return (
12       <BrowserRouter>
13         <div className="App">
14           <Navbar />
15           <Route exact path='/' component={Home} />
16           <Route path='/about' component={About} />
17           <Route path='/contact' component={Contact} />
18           <Route path="/posts/:post_id" component={Post} />
19         </div>
20       </BrowserRouter>
21     );
22   }
23 }
```

The screenshot shows the Visual Studio Code interface with the following details:

- Title Bar:** Home.js - react-redux-complete - Visual Studio Code
- Left Sidebar (Explorer):** Shows the project structure with files like Home.js, App.js, Post.js, and components under the poketimes folder.
- Right Side Panel:** Shows a tree view labeled "REACT-REDUX-COMPLETE" with components like node_modules, public, and src.
- Editor Area:** Displays the code for Home.js. The code uses axios to fetch data from a JSONPlaceholder API endpoint and maps it to a list of posts.

```
10 axios.get('https://jsonplaceholder.typicode.com/posts')
11 .then(res => {
12   console.log(res)
13   this.setState({
14     posts: res.data.slice(0,10)
15   })
16 }
17 }
18 render() {
19   const { posts } = this.state;
20   const postList = posts.length ? (
21     posts.map(post => {
22       return (
23         <div className="post card" key={post.id}>
24           <div className="card-content">
25             <Link to={`/posts`} + post.id>
26               <span className="card-title">{post.title}</span>
27             </Link>
28             <p>{post.body}</p>
29           </div>
30         </div>
31       )
32     })
33   )
34 }
```



The screenshot shows a Visual Studio Code interface with the following details:

- Title Bar:** App.js - react-redux-complete - Visual Studio Code
- Explorer View (Left):** Shows the project structure:
 - OPEN EDITORS: Home.js, App.js, Post.js
 - REACT-REDUX-COMPLETE folder
 - poketimes
 - node_modules
 - public
 - src
 - components
 - About.js
 - Contact.js
 - Home.js
 - Navbar.js
 - Post.js
 - hoc
 - Rainbow.js
 - App.js
 - index.css, index.js, registerServiceWorker.js
 - .gitignore, package-lock.json, package.json
 - README.md
- Code Editor (Right):** Displays the content of the App.js file:

```
1 import React, { Component } from 'react';
2 import Navbar from './components/Navbar'
3 import { BrowserRouter, Route, Switch } from 'react-router-dom'
4 import Home from './components/Home'
5 import Contact from './components/Contact'
6 import About from './components/About'
7 import Post from './components/Post'

8 class App extends Component {
9   render() {
10     return (
11       <BrowserRouter>
12         <div className="App">
13           <Navbar />
14           <Switch>
15             <Route exact path='/' component={Home} />
16             <Route path='/about' component={About} />
17             <Route path='/contact' component={Contact} />
18             <Route path="/:post_id" component={Post} />
19           </Switch>
20         </div>
21       </BrowserRouter>
22     );
23   }
}
```

IMPORTING IMAGES

A screenshot of the Visual Studio Code interface showing a React-Redux application named "Home.js".

The Explorer sidebar on the left shows the project structure:

- OPEN EDITORS: 1 UNSAVED
- Home.js (selected)
- REACT-REDUX-COMPLETE
- poketimes
- node_modules
- public
- src
 - components
 - hoc
 - App.js
 - index.css
 - index.js
 - pokeball.png
 - registerServiceWorker.js
- .gitignore
- package-lock.json
- package.json
- README.md

The main editor area displays the Home.js file content:

```
1 import React, { Component } from 'react'
2 import axios from 'axios'
3 import { Link } from 'react-router-dom'
4 import Pokeball from '../pokeball.png'
5
6 class Home extends Component {
7   state = {
8     posts: []
9   }
10  componentDidMount(){
11    axios.get('https://jsonplaceholder.typicode.com/posts')
12      .then(res => {
13        console.log(res)
14        this.setState({
15          posts: res.data.slice(0,10)
16        })
17      })
18  }
19  render() {
20    const { posts } = this.state;
21    const postList = posts.length ? (
22      posts.map(post => {
```

A screenshot of the Visual Studio Code interface. The title bar says "Home.js - react-redux-complete - Visual Studio Code". The left sidebar shows the project structure in the Explorer tab:

- OPEN EDITORS
- REACT-REDUX-COMPLETE
- poketimes
- node_modules
- public
- src
 - components
 - hoc
 - App.js
 - index.css
 - index.js
 - pokeball.png
 - registerServiceWorker.js
- .gitignore
- package-lock.json
- package.json
- README.md

The main editor area displays the code for Home.js:

```
10  componentDidMount(){
11    axios.get('https://jsonplaceholder.typicode.com/posts')
12      .then(res => {
13        console.log(res)
14        this.setState({
15          posts: res.data.slice(0,10)
16        })
17      })
18    }
19    render() {
20      const { posts } = this.state;
21      const postList = posts.length ? (
22        posts.map(post => {
23          return (
24            <div className="post card" key={post.id}>
25              <img src={Pokeball} alt="A pokeball"/>
26              <div className="card-content">
27                <Link to={'/' + post.id}>
28                  <span className="card-title">{post.title}</span>
29                </Link>
30                <p>{post.body}</p>
31              </div>
32            </div>
33          )
34        )
35      ) : <div>No posts available</div>;
36      return (
37        <div>
38          <h1>Poketimer</h1>
39          <h2>Recent Posts</h2>
40          {postList}
41        </div>
42      )
43    }
44  }
45}
```

React App

localhost:3001

Poke'Times

Home About Contact

Home



sunt aut facere repellat provident occaecati excepturi optio
reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut ut quas
totam nostrum rerum est autem sunt rem eveniet architecto

index.css - react-redux-complete - Visual Studio Code

EXPLORER OPEN EDITORS 1 UNSAVED Home.js index.css

REACT-REDUX-COMPLETE

poketimes

node_modules

public

src

components

About.js

Contact.js

Home.js

Navbar.js

Post.js

hoc

App.js

index.css

index.js

pokeball.png

registerServiceWorker.js

.gitignore

package-lock.json

package.json

README.md

```
1 body {  
2   margin: 0;  
3   padding: 0;  
4   font-family: sans-serif;  
5 }  
6  
7 .home .post img{  
8   position: absolute;  
9   top: 20px;  
10  left: -100px;  
11  opacity: 0.6;  
12 }  
13  
14 .home .post{  
15   overflow: hidden;  
16   padding-left: 80px  
17 }  
18
```

A screenshot of a web browser window displaying a website titled "Poke'Times". The browser's address bar shows "localhost:3000". The page has a red header bar with the title "Poke'Times" and navigation links for "Home", "About", and "Contact". The main content area has a light gray background. It features two cards. The first card, titled "Home", contains a small icon of a Poké Ball on the left and text in blue and black. The second card, titled "qui est esse", also contains a small icon of a Poké Ball on the left and text in blue and black. The text in the cards appears to be placeholder Latin text.

Poke'Times

Home About Contact

Home

sunt aut facere repellat provident occaecati excepturi
optio reprehenderit

quia et suscipit suscipit recusandae consequuntur expedita et cum reprehenderit molestiae ut
ut quas totam nostrum rerum est autem sunt rem eveniet architecto

qui est esse

est rerum tempore vitae sequi sint nihil reprehenderit dolor beatae ea dolores neque fugiat
blanditiis voluptate porro vel nihil molestiae ut reiciendis qui aperiam non debitis possimus qui
neque nisi nulla