

AJAX と API

AJAX コールをする方法は？

任意の AJAX ライブラリを React と共に利用可能です。人気のあるものとしては、[Axios](#)、[jQuery AJAX](#)、ブラウザ組み込みの [window.fetch](#) などがあります。

コンポーネントのどのライフサイクルで AJAX コールすべきか？

AJAX コールによるデータ取得は `componentDidMount` のライフサイクルメソッドで行うべきです。データ取得後に `setState` でコンポーネントを更新できるようにするためです。

例：ローカル state に AJAX の通信結果をセットする

下記のコンポーネントは、`componentDidMount` で AJAX コールして得られたデータをローカルコンポーネントの state に流し込んでいます。

このサンプル API が返す JSON オブジェクトはこのようになります：

```
{
  "items": [
    { "id": 1, "name": "Apples", "price": "$2" },
    { "id": 2, "name": "Peaches", "price": "$5" }
  ]
}
```

```
class MyComponent extends React.Component {
  constructor(props) {
    super(props);
    this.state = {
      error: null,
      isLoading: false,
      items: []
    };
  }

  componentDidMount() {
    fetch("https://api.example.com/items")
      .then(res => res.json())
      .then(
        (result) => {
          this.setState({
            isLoading: true,
            items: result.items
          });
        },
        // 補足：コンポーネント内のバグによる例外を隠蔽しないためにも
        // catch() ブロックの代わりにここでエラーハンドリングすることが重要です
        (error) => {
          this.setState({
            isLoading: true,
            error
          });
        }
      )
  }

  render() {
    const { error, isLoading, items } = this.state;
    if (error) {
      return <div>Error: {error.message}</div>;
    } else if (!isLoading) {
      return <div>Loading...</div>;
    } else {
      return (
        <ul>
          {items.map(item => (
            <li key={item.name}>
              {item.name} {item.price}
            </li>
          ))}
        </ul>
      );
    }
  }
}
```

[このページを編集する](#)

