

CONTEXT APIを使っ  
たライブラリを作った

最初に

# 自己紹介

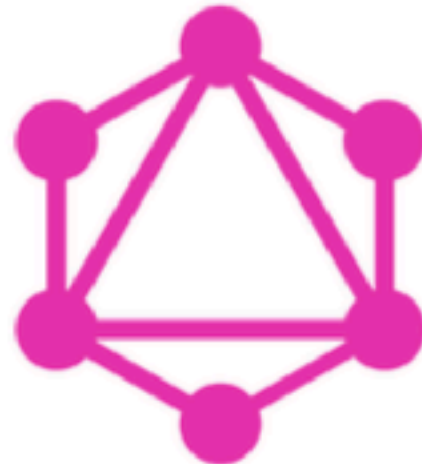
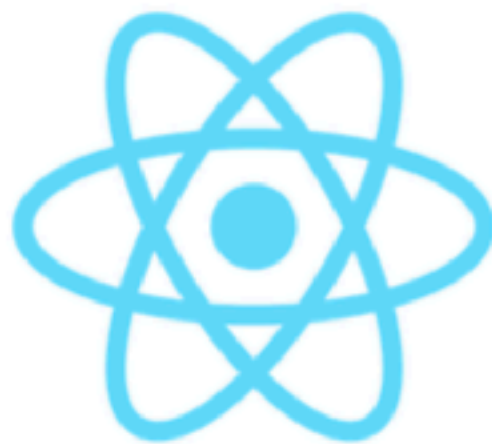
- 加藤大志 フリーランスプログラマー



11月  
7

【初参加歓迎】JavaScriptを学ぶReact/GraphQLアプリ  
開発塾 #42 ★毎週水曜夜

毎週水曜の夜に開催するJavaScript講座



アプリ開発塾



# JAVASCRIPTコーディングについて

- コーディングは好きだけど、できるだけ少なく書きたい
- アロー関数に統一、functionもclassも書かない
- this-less JavaScript
- 副作用ない関数好みなので、reducer/reduxは好み

# REDUX好きだけど

- 関数合成でロジックを組み立てるコンセプトは好きだけど
- コーディングが増えるのは悩み
- 簡単そうに見えて実は初心者に優しくない
- なんとなくデファクトっぽいけど、小規模開発にはオーバースペック

さて REACT



# REACT 16.3～

- 公式のContext API
  - context自体は以前からあったが、通常用途では使うと言われていた
- render props (children) の公式採用
  - HoCの方がクリーンな気はするけど、直感的分かりやすさや変数スコープは良い

## REDUX代替？

- Context APIでReduxの代替ができるとの噂
- connectにしかメリットを感じないケースはそうかもしれない
- それはそもそもReduxは必要なかったケース
- 余談：connectを使わずにReduxを使うこともできる（reducerにメリット、パフォーマンス気にしない）



# CONTEXT APIの使い方

```
1  const ThemeContext = React.createContext('default');
2
3  const App = () => (
4    <ThemeContext.Provider value={'dark'}>
5      <ToolBar />
6      { /* ... */ }
7    </ThemeContext.Provider>
8  );
9
10 const ToolBar = () => (
11   <MenuBar>
12     <HomeButton />
13     { /* ... */ }
14   </MenuBar>
15 );
16
17 const HomeButton = () => (
18   <ThemeContext.Consumer>
19     {value => (
20       <Button theme={value}>Home</Button>
21     )}
22   </ThemeContext.Consumer>
23 );
```

## CONTEXT APIのおさらい

- コンポーネントツリーの階層をジャンプして、値を渡すことができる
- Providerで値を渡して、Consumerで値を受け取る
- 一種のグローバル変数なので乱用は禁物
- コンポーネントの再利用性が下がる

# CONTEXT APIでCONNECTの代替

- コンポーネントツリーをジャンプできるが、Context APIだけではstate管理ができるわけではない
- stateを更新したら、必要なコンポーネントだけ再レンダリングするようになって欲しい
- サンプルコードがちらほら



# CONTEXT APIでGLOBAL STATE

```
1  const DEFAULT_STATE = { email: 'unknown' };
2
3  const { Provider, Consumer } = React.createContext(DEFAULT_STATE);
4
5  class App extends React.Component {
6    state = DEFAULT_STATE;
7    setEmail = email => {
8      this.setState({ email });
9    };
10   render() {
11     return (
12       <Provider value={{ ...this.state, setEmail: this.setEmail }}>
13         <Viewer />
14         <Editor />
15       </Provider>
16     );
17   }
18 }
```

# CONTEXT APIでGLOBAL STATE

```
20  const Viewer = () => (  
21    <Consumer>  
22      |    ({ email }) => (  
23        |    <ShowEmail>{email}</ShowEmail>  
24        |    )}  
25    </Consumer>  
26  );  
27  
28  const Editor = () => (  
29    <Consumer>  
30      |    ({ email, setEmail }) => (  
31        |    <EmailField  
32        |      value={email}  
33        |      onUpdate={newValue => setEmail(newValue)}  
34        |    />  
35      |    )}  
36    </Consumer>  
37  );
```

あ！

CLASS使っちゃった



## CONTEXT APIのおさらい

- Providerはvalueが変更したら、対となる(複数の)Consumerを再レンダリングする
- valueが変更されなければ何も起こらない
- valueを変更するためにComponent.setStateを使わないといけない

それは嫌～

# ライブラリにした



🔍 Search packages

## react-context-global-state

0.6.0 • **Public** • Published 20 hours ago

[Readme](#)

0 Dependencies

## react-context-global-state

build **passing** npm package **0.6.0**

Simple global state for React by Context API

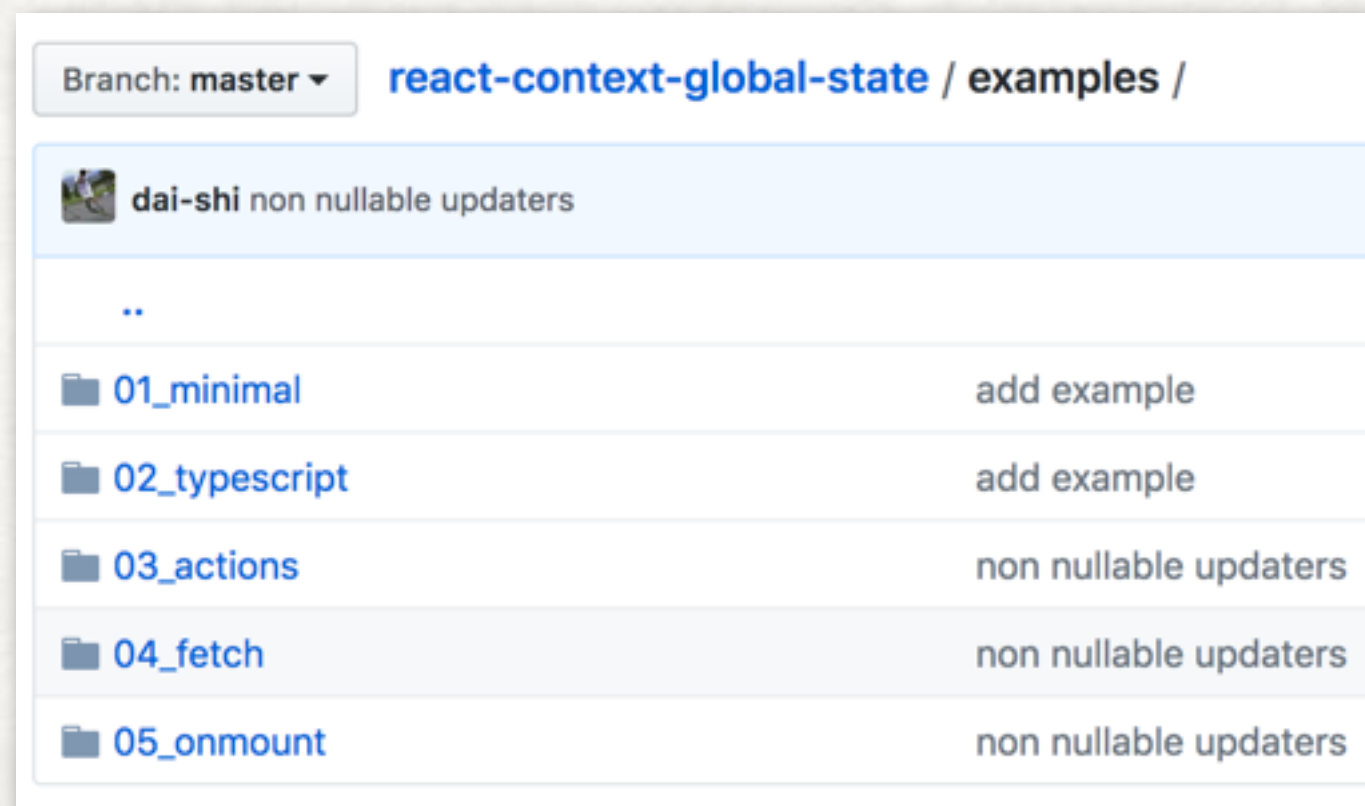


# 改めてGLOBAL STATE

```
1  const { StateProvider, StateConsumer } = createGlobalState({ email: 'unknown' });
2
3  const App = () => (
4    <StateProvider>
5      <Viewer />
6      <Editor />
7    </StateProvider>
8  );
9
10 const Viewer = () => (
11   <StateConsumer name="email">
12     {email => (
13       <ShowEmail>{email}</ShowEmail>
14     )}
15   </StateConsumer>
16 );
17
18 const Editor = () => (
19   <StateConsumer name="email">
20     {(email, setEmail) => (
21       <EmailField
22         value={email}
23         onUpdate={newValue => setEmail(newValue)}
24       />
25     )}
26   </StateConsumer>
27 );
```

# REACT-CONTEXT-GLOBAL-STATEおさらい

- 特徴1: setStateを使って再レンダリングする
- 特徴2: 複数のcontextを一つにまとめてくれる
- 特徴3: TypeScript用の型定義がされている



おまけ



# REACT 16.7

- React Hooks APIで世界が変わる
- Function Components (statelessだけじゃない)
- classでしかできなかったことがなくなる
- render propsで書かなくてもよくなる
- custom hooksがいっぱい登場する

# 実験段階のライブラリ



🔍 Search packages

Share your code. npm Orgs help your team discover, share

## react-hooks-global-state

0.0.1 • **Public** • Published 18 hours ago

[Readme](#)

0 Dependencies

## react-hooks-global-state

build **passing** npm package **0.0.1**

Simple global state for React by Hooks API

**THANK YOU!**