

React.js 小书

[<-- 返回首页](#)

使用真正的 Redux 和 React-redux

- 作者: [胡子大哈](#)
- 原文链接: <http://huziketang.com/books/react/lesson42>
- 转载请注明出处, 保留原文链接和作者信息。

(本文未审核)

现在 `make-react-redux` 工程代码中的 Redux 和 React-redux 都是我们自己写的, 现在让我们来使用真正的官方版本的 Redux 和 React-redux。

在工程目录下使用 npm 安装 Redux 和 React-redux 模块:

```
npm install redux react-redux --save
```

把 `src/` 目录下 `Header.js`、`ThemeSwitch.js`、`Content.js` 的模块导入中的:

```
import { connect } from './react-redux'
```

改成:

```
import { connect } from 'react-redux'
```

也就是本来从本地 `./react-redux` 导入的 `connect` 改成从第三方 `react-redux` 模块中导入。

修改 `src/index.js`, 把前面部分的代码调整为:

```
import React, { Component } from 'react'
import ReactDOM from 'react-dom'
import { createStore } from 'redux'
import { Provider } from 'react-redux'
import Header from './Header'
import Content from './Content'
import './index.css'

const themeReducer = (state, action) => {
  if (!state) return {
    themeColor: 'red'
  }
  switch (action.type) {
```

```
    case 'CHANGE_COLOR':
      return { ...state, themeColor: action.themeColor }
    default:
      return state
  }
}

const store = createStore(themeReducer)

...
```

我们删除了自己写的 `createStore`，改成使用第三方模块 `redux` 的 `createStore`；`Provider` 本来从本地的 `./react-redux` 引入，改成从第三方 `react-redux` 模块中引入。其余代码保持不变。

接着删除 `src/react-redux.js`，它的已经用处不大了。最后启动工程 `npm start`：



可以看到我们原来的业务代码其实都没有太多的改动，实际上我们实现的 `redux` 和 `react-redux` 和官方版本在该场景的用法上是兼容的。接下来的章节我们都会使用官方版本的 `redux` 和 `react-redux`。

课后练习

- [React-redux 实现用户列表的显示、增加、删除](#)

因为第三方评论工具有问题，对本章节有任何疑问的朋友可以移步到 [React.js 小书的论坛](#) 发帖，我会回答大家的疑问。

下一节：[Smart 组件 vs Dumb 组件](#)

上一节：[动手实现 React-redux（六）：React-redux 总结](#)

如果你觉得小书写得还不错，可以请胡子大哈喝杯茶 :)

赞赏

或者传播一下知识也是一个很好的选择

2 条评论，2 人参与。

★ 2



我有话说...

使用社交帐号登录

发布前先点击左边的按钮登录

最新评论



神木 • 6月16日 17:45
看东西不说话
顶 • 回复 • 分享»



似梦飞花 • 3月31日 18:28
赞 写的非常好
顶 • 回复 • 分享»

友言?