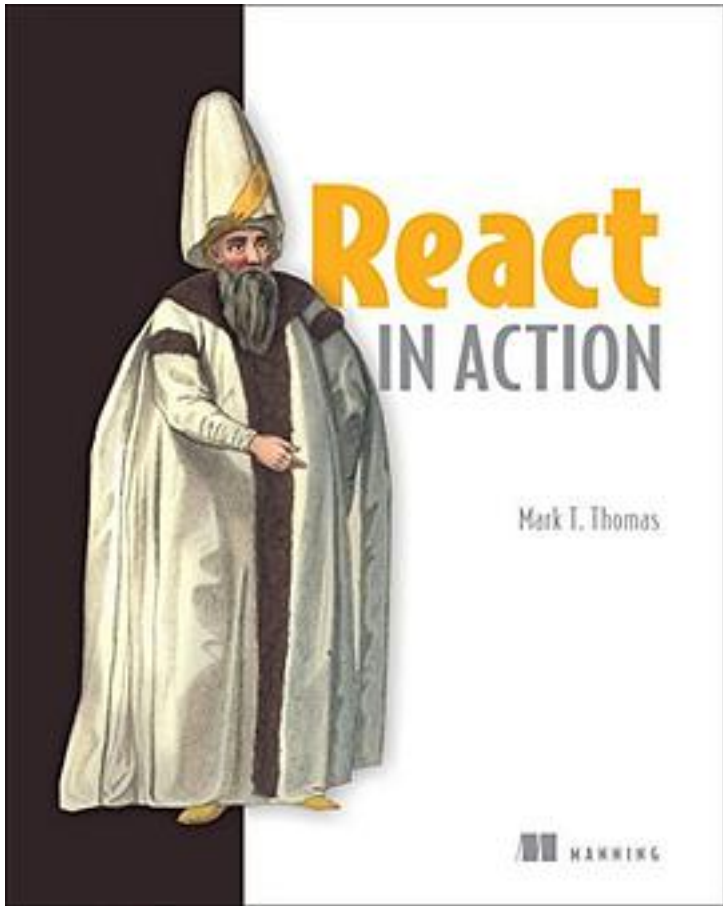


React in Action



[React in Action_ 下载链接1](#)

著者:Mark Tielens Thomas

出版者:Manning Publications

出版时间:2017-7-1

装帧:Paperback

isbn:9781617293856

React in Action introduces front-end developers to the React framework and related tools. This clearly written, example-rich book begins by introducing you to React, diving into some of the fundamental ideas in React, and working with components. In the second section, you'll explore the different ways that data works in React as well as

learning more about components. You'll also find several useful appendixes covering related topics like React tooling and the React ecosystem.

About the Technology

Facebook created React to help deliver amazing user experiences on a website with thousands of components and an incomprehensible amount of traffic. The same powerful tools are available to you too! The key is a clever design for managing state, data flow, and rendering, so your application is easy to think about and runs smoothly. Add an incredibly rich ecosystem of components and libraries, and you've got a recipe for building web apps that will delight both developers and users.

React in Action teaches you to think like a pro about user interfaces and building them with React. This practical book gets you up and running quickly with hands-on examples in every chapter. You'll master core topics like rendering, lifecycle methods, JSX, data flow, forms, routing, integrating with third-party libraries, and testing. And the included application design ideas will help make your apps pop. As you learn to integrate React into full-stack applications, you'll explore state management with Redux and server-side rendering, and even dabble in React Native for mobile UIs.

What's inside

React from the ground up

Implementing a routing system with components

Server-side rendering in Node.js

Working with third-party libraries

Testing React components

作者介绍:

马克·蒂伦斯·托马斯 (Mark Tielens Thomas): 一位经验丰富的软件工程师, 他每天都在用React、JavaScript和Node.js工作。他喜爱整洁的代码、优美的系统和上好的咖啡。

任发科: 火币高级研发总监, 曾任职亚马逊、唯品会等多家互联网公司, 担任研发和技术管理工作, 有丰富的软件架构、开发和管理经验。个人长期从事和关注高效研发组织的构建和管理, 并有丰富的团队管理实践。近年主要关注和从事研发效能和DevOps体系的建立, 目前从事稳定性工程的相关工作。

陈伟: 哗啦啦前端架构师。曾在唯品会、火币等公司任前端工程师和前端架构师。深入理解JavaScript语言以及Node.js、Vue、React等前端框架, 并在前端组件化方向有深入的工程化研究。目前致力于可视化的页面编辑器的设计和开发, 赋能产品与前端, 提升公司开发效率。

蒋峰: 火币网资深前端工程师, 曾就职于国家农业信息化中心、阿里健康、融数金服等从事软件研发、基础架构等工作, 目前主要负责火币网前端相关产品研发管理工作。早年间致力于微软.NET框架研发, 近年专注于前端研发体系架构, 有大量的C#、Node.js和JavaScript项目开发经验, 对Electron跨平台应用有极大的兴趣与研究。目前正在积极推进企业内中后台微前端应用方案的验证与实施。

邱巍：现就职于哗啦啦研发中心，曾就职于作业盒子、融数金服、火币网等从事软件研发工作，目前主要负责数据可视化基础组件研发。对图表绘制、图表交互、数据建模与分析等相关内容有极大兴趣。目前致力于提供展现更准确，分析更高效的数据可视化工具研发。

目录: List of Figures

List of Tables

List of Listings

Table of Contents

Preface

Acknowledgments

About this book

About the author

About the cover illustration

第一部分 Meet React

Ch 1. Meet React

1.1. Meet React

1.1.1. Who this book is for

1.1.2. A note on tooling

1.1.3. Who uses React?

1.2. What does React not do?

1.2.1. Tradeoffs of React

1.3. The virtual DOM

1.3.1. The DOM

1.3.2. The virtual DOM

1.3.3. Updates and diffing

1.3.4. Virtual DOM: Need for speed?

1.4. Components: The fundamental unit of React

1.4.1. Components in general

1.4.2. Components in React: Encapsulated and reusable

1.5. React for teams

1.6. Summary

Ch 2. <Hello World />: our first component

2.1. Introducing React components

2.1.1. Understanding the application data

2.1.2. Multiple components: Composition and parent-child relationships

2.1.3. Establishing component relationships

2.2. Creating components in React

2.2.1. Creating React elements

2.2.2. Rendering your first component

2.2.3. Creating React components

2.2.4. Creating React classes

2.2.5. The render method

2.2.6. Property validation via PropTypes

2.3. The life and times of a component

2.3.1. A React state of mind

2.3.2. Setting initial state

2.4. Meet JSX

2.4.1. Creating components using JSX

2.4.2. Benefits of JSX and differences from HTML

2.5. Summary

第二部分 Components and data in React

Ch 3. Data and data flow in React

- 3.1. Introducing state
 - 3.1.1. What is state?
 - 3.1.2. Mutable and immutable state
- 3.2. State in React
 - 3.2.1. Mutable state in React: Component state
 - 3.2.2. Immutable state in React: Props
 - 3.2.3. Working with props: PropTypes and default props
 - 3.2.4. Stateless functional components
- 3.3. Component communication
- 3.4. One-way data flow
- 3.5. Summary
- Ch 4. Rendering and lifecycle methods in React
 - 4.1. Getting set up with the Letters Social repo
 - 4.1.1. Getting the source code
 - 4.1.2. Which version of node should I use?
 - 4.1.3. Note on tooling and CSS
 - 4.1.4. Deploying
 - 4.1.5. The API server and database
 - 4.1.6. Running the app
 - 4.2. The render process and lifecycle methods
 - 4.2.1. Introducing lifecycle methods
 - 4.2.2. Types of lifecycle methods
 - 4.2.3. Initial and “will” methods
 - 4.2.4. Mounting components
 - 4.2.5. Updating methods
 - 4.2.6. Unmounting methods
 - 4.2.7. Catching errors
 - 4.3. Starting to create Letters Social
 - 4.4. Summary
- Ch 5. Working with forms in React
 - 5.1. Creating posts in Letters Social
 - 5.1.1. Data requirements
 - 5.1.2. Component overview and hierarchy
 - 5.2. Forms in React
 - 5.2.1. Getting started with forms
 - 5.2.2. Form elements and events
 - 5.2.3. Updating state in forms
 - 5.2.4. Controlled and uncontrolled components
 - 5.2.5. Form validation and sanitization
 - 5.3. Creating new posts
 - 5.4. Summary
- Ch 6. Integrating third-party libraries with React
 - 6.1. Sending posts to the Letters Social API
 - 6.2. Enhancing your component with maps
 - 6.2.1. Creating the DisplayMap component using refs
 - 6.2.2. Creating the LocationTypeAhead component
 - 6.2.3. Updating CreatePost and adding maps to posts
 - 6.3. Summary
- Ch 7. Routing in React
 - 7.1. What is routing?
 - 7.1.1. Routing in modern front-end web applications
 - 7.2. Creating a router
 - 7.2.1. Component routing
 - 7.2.2. Creating the <Route /> component

- 7.2.3. Starting to build the <Router/> component
- 7.2.4. Matching URL paths and parameterized routing
- 7.2.5. Adding routes to the Router component
- 7.3. Summary
- Ch 8. More routing and integrating Firebase
 - 8.1. Using the router
 - 8.1.1. Creating a page for a post
 - 8.1.2. Creating a <Link/> component
 - 8.1.3. Creating a <NotFound/> component
 - 8.2. Integrating Firebase
 - 8.2.1. Ensuring a user is logged in
 - 8.3. Summary
- Ch 9. Testing React components free
 - 9.1. Types of testing
 - 9.1.1. Why test?
 - 9.2. Testing React components with Jest, Enzyme, and React-test-renderer
 - 9.3. Writing your first tests
 - 9.3.1. Getting started with Jest
 - 9.3.2. Testing a stateless functional component
 - 9.3.3. Testing the CreatePost component without Enzyme
 - 9.3.4. Test coverage
 - 9.4. Summary
- 第三部分 React Application architecture
- Ch 10. Redux application architecture
 - 10.1. The Flux application architecture
 - 10.1.1. Meet Redux: A variation on Flux
 - 10.1.2. Getting set up for Redux
 - 10.2. Creating actions in Redux
 - 10.2.1. Defining action types
 - 10.2.2. Creating actions in Redux
 - 10.2.3. Creating the Redux store and dispatching actions
 - 10.2.4. Asynchronous actions and middleware
 - 10.2.5. To Redux or not to Redux?
 - 10.2.6. Testing actions
 - 10.2.7. Creating custom Redux middleware for crash reporting
 - 10.3. Summary
- Ch 11. More Redux and integrating Redux with React
 - 11.1. Reducers determine how state should change
 - 11.1.1. State shape and initial state
 - 11.1.2. Setting up reducers to respond to incoming action
 - 11.1.3. Combining reducers together in our store
 - 11.1.4. Testing reducers
 - 11.2. Bringing React and Redux together
 - 11.2.1. Containers vs. presentational components
 - 11.2.2. Using <Provider /> to connect components to the Redux store
 - 11.2.3. Binding actions to component event handlers
 - 11.2.4. Updating your tests
 - 11.3. Summary
- Ch 12. React on the server and integrating React Router
 - 12.1. What is server-side rendering?
 - 12.1.1. Digging into server-side rendering
 - 12.2. Why render on the server?
 - 12.3. You might not need SSR
 - 12.4. Rendering components on the server

- 12.5. Switching to React Router
 - 12.5.1. Setting up React router
- 12.6. Handling authenticated routes with React router
- 12.7. Server rendering with data-fetching
- 12.8. Summary
- Ch 13. An introduction to React Native
 - 13.1. Introducing React Native
 - 13.2. React and React Native
 - 13.3. When to use React Native
 - 13.4. The simplest “Hello World”
 - 13.5. Where to go next
 - 13.6. Summary
- • • • • (收起)

[React in Action 下载链接1](#)

标签

计算机

重要度./.×××

重要度./.×

软件开发

计算机科学

前端技术

主题./.工作./.JS

React

评论

内容很详实

[React in Action_下载链接1_](#)

书评

[React in Action_下载链接1_](#)