

React.js 小书

[<-- 返回首页](#)

更新阶段的组件生命周期

- 作者: [胡子大哈](#)
- 原文链接: <http://huziketang.com/books/react/lesson20>
- 转载请注明出处, 保留原文链接和作者信息。

从之前的章节我们了解到, 组件的挂载指的是将组件渲染并且构造 DOM 元素然后插入页面的过程。这是一个从无到有的过程, React.js 提供一些生命周期函数可以给我们在这个过程中做一些操作。

除了挂载阶段, 还有一种“更新阶段”。说白了就是 `setState` 导致 React.js 重新渲染组件并且把组件的变化应用到 DOM 元素上的过程, 这是一个组件的变化过程。而 React.js 也提供了一系列的生命周期函数可以让我们在这个组件更新的过程执行一些操作。

这些生命周期在深入项目开发阶段是非常重要的。而要完全理解更新阶段的组件生命周期是一个需要经验和知识积累的过程, 你需要对 Virtual-DOM 策略有比较深入理解才能完全掌握, 但这超出了本书的目的。本书的目的是为了让大家快速掌握 **React.js** 核心的概念, 快速上手项目进行实战。所以对于组件更新阶段的组件生命周期, 我们简单提及并且提供一些资料给大家。

这里为了知识的完整, 补充关于更新阶段的组件生命周期:

1. `shouldComponentUpdate(nextProps, nextState)`: 你可以通过这个方法控制组件是否重新渲染。如果返回 `false` 组件就不会重新渲染。这个生命周期在 React.js 性能优化上非常有用。
2. `componentWillReceiveProps(nextProps)`: 组件从父组件接收到新的 `props` 之前调用。
3. `componentWillUpdate()`: 组件开始重新渲染之前调用。
4. `componentDidUpdate()`: 组件重新渲染并且把更改变更到真实的 DOM 以后调用。

大家对这更新阶段的生命周期比较感兴趣的话可以查看[官网文档](#)。

但这里建议大家可以先简单了解 **React.js** 组件是有更新阶段的, 并且有这么几个更新阶段的生命周期即可。然后在深入项目实战的时候逐渐地掌握理解他们, 现在并不需要对他们放过多的精力。

有朋友对 Virtual-DOM 策略比较感兴趣的话, 可以参考这篇博客: [深度剖析: 如何实现一个 Virtual DOM 算法](#)。对深入理解 React.js 核心算法有一定帮助。

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：[ref 和 React.js 中的 DOM 操作](#)

上一节：[挂载阶段的组件生命周期（二）](#)

如果你觉得小书写得还不错，可以请胡子大哈喝茶 :)

赞赏

或者传播一下知识也是一个很好的选择

1 条评论，1 人参与。

★ 0



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



阳光老男孩 • 4月5日 04:24

"也提供了一些列的生命周期函数", "一些列"应为"一系列"。感谢作者的辛苦贡献，这真是一本清晰明了的react入门小书，娓娓道来，感觉学习曲线降了好多

顶 • 回复 • 分享»

友言?