



链滴

React 从零到一 Demo 演练 (下)

作者: [sirwsl](#)

原文链接: <https://ld246.com/article/1621998635460>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

废话

这一部分比前一篇难懂些，但是还是的了解、掌握，至于用嘛，后面自己在好好研究，由于写这套教时候我才学习一个周，所以可能有错的地方，多包涵。

React API

React的Api有很多，这里为了后面AntD的上手，就简单介绍6个，够用就行，分别是：

- 设置状态：setState
- 替换状态：replaceState
- 设置属性：setProps
- 替换属性：replaceProps
- 强制更新：forceUpdate

我们——来看一下：

setState

在之前的样例中有简单的使用，我们再来看一下：

```
setState(object nextState[, function callback])
```

eg:

```
this.setState({count : this.state.count+1},()=>{  
  //回调需要处理的业务逻辑  
})
```

大概就是这样，但是需要注意：**callback回调函数必须是state设置成功且组件渲染完毕才会触发**

replaceState

该方法会替代当前的state，然后设置新的state，原有的state会被全部删除，后面也是回调函数

这个在之前的demo中没有出现过，但是方法基本也是一样的

```
replaceState(object nextState[, function callback])
```

eg:

```
this.replaceState({num:0},()=>{  
  //取代成功后，渲染结束触发回调函数  
})
```

setProps

在上一篇博客说到：**props** 相当于组件的数据流，它总是会从父组件向下传递至所有的子组件中

这里的setProps使用与state类似，但是setProps会设置新的属性并与当前属性合并

```
setProps(object nextProps[, function callback])
```

replaceProps

让设置的新属性，取代原有的属性

```
setProps(object nextProps[, function callback])
```

forceUpdate

强制更新渲染，该方法会在render()方法执行后调用

```
forceUpdate([function callback])
```

eg:

```
this.forceUpdate(()=>{  
  //强制更新操作  
})
```

React生命周期

react生命周期，这部分可能我写的会比较垃圾，建议看看其他人写的

这里推荐：

[详解React生命周期\(包括react16最新版\)](#)

[React - 组件的生命周期详解（及各阶段调用的方法）](#)

[React组件生命周期详解](#)

看了对应的博客后应该就大概知道了这个东西的存在

简单说一下：

- **componentWillMount** 组件在被调用时触发，也就是渲染之前，可根据业务处理state，也是在render调用前处理state的唯一机会。
- **componentDidMount**：在第一次渲染后调用，会被触发，一般我们请求外部数据接口放这里面当组件被挂后只会被触发一次
- **componentWillReceiveProps** 在组件接收到一个新的 prop (更新后)时被调用。这个方法在初始render时不会被调用。
- **shouldComponentUpdate** 返回一个布尔值。在组件接收到新的props或者state时被调用。在初始化时或者使用forceUpdate时不被调用。可以在你确认不需要更新组件时使用。
- **componentWillUpdate** 在组件接收到新的props或者state但还没有render时被调用。在初始化不会被调用。
- **componentDidUpdate** 在组件完成更新后立即调用。在初始化时不会被调用。
- **componentWillUnmount** 在组件从 DOM 中移除之前立刻被调用。

eg:

```
class Child extends Component {
  constructor(props) {
    super(props);
    this.state = {
      someThings: props.someThings
    };
  }
  componentWillReceiveProps(nextProps) { // 父组件重传props时就会调用这个方法
    this.setState({someThings: nextProps.someThings});
  }
  render() {
    return <div>{this.state.someThings}</div>
  }
}
```

学到这里基本能够开始去看antD的代码了

相关链接

上一篇: [eyes](#)

[React从零到一Demo演练 \(上\)](#)

下一篇: [eyes](#)

[Ant Design Pro从零到一 \(认识AntD\)](#)

Ant Design Pro 系列: [eyes](#)

[Ant Design Pro 从零到一教程](#)

[React从零到一Demo演练 \(上\)](#)

[React从零到一Demo演练 \(下\)](#)

[Ant Design Pro从零到一 \(认识AntD\)](#)

[Ant Design Pro从零到一 \(页面创建\)](#)

[Ant Design Pro从零到一 \(Mock使用\)](#)

[Ant Design Pro 从零到一 \(学习Model\)](#)

[Ant Design Pro 从零到一 \(总结\)](#)