

React.js 小书

[<-- 返回首页](#)

## dangerouslySetHTML 和 style 属性

- 作者: [胡子大哈](#)
- 原文链接: <http://huziketang.com/books/react/lesson23>
- 转载请注明出处, 保留原文链接和作者信息。

这一节我们来补充两个之前没有提到的属性, 但是在 React.js 组件开发中也非常常用, 但是它们也很简单。

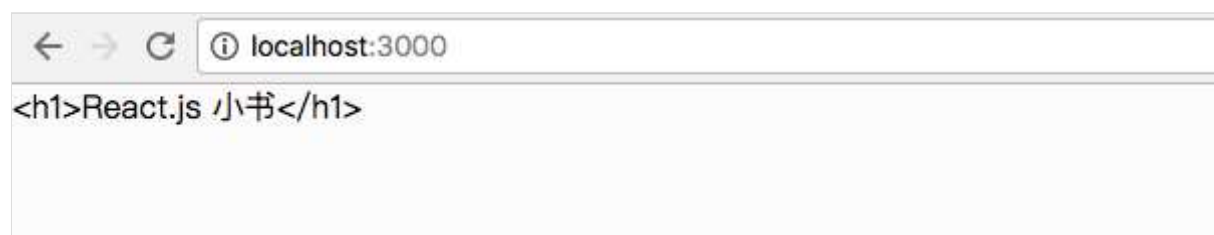
### dangerouslySetHTML

出于安全考虑的原因 (XSS 攻击), 在 React.js 当中所有的表达式插入的内容都会被自动转义, 就相当于 jQuery 里面的 `text(...)` 函数一样, 任何的 HTML 格式都会被转义掉:

```
class Editor extends Component {
  constructor() {
    super()
    this.state = {
      content: '<h1>React.js 小书</h1>'
    }
  }

  render () {
    return (
      <div className='editor-wrapper'>
        {this.state.content}
      </div>
    )
  }
}
```

假设上面是一个富文本编辑器组件, 富文本编辑器的内容是动态的 HTML 内容, 用 `this.state.content` 来保存。我希望在编辑器内部显示这个动态 HTML 结构, 但是因为 React.js 的转义特性, 页面上会显示:



表达式插入并不会把一个 `<h1>` 渲染到页面，而是把它的文本形式渲染了。那要怎样才能做到设置动态 HTML 结构的效果呢？React.js 提供了一个属性

`dangerouslySetInnerHTML`，可以让我们设置动态设置元素的 `innerHTML`：

```
...
render () {
  return (
    <div
      className='editor-wrapper'
      dangerouslySetInnerHTML={{__html: this.state.content}} />
  )
}
...
```

需要给 `dangerouslySetInnerHTML` 传入一个对象，这个对象的 `__html` 属性值就相当于元素的 `innerHTML`，这样我们就可以动态渲染元素的 `innerHTML` 结构了。

有写朋友会觉得很奇怪，为什么要把一件这么简单的事情搞得这么复杂，名字又长，还要传入一个奇怪的对象。那是因为设置 `innerHTML` 可能会导致跨站脚本攻击（XSS），所以 React.js 团队认为把事情搞复杂可以防止（警示）大家滥用这个属性。这个属性不必要的情况就不要使用。

## style

React.js 中的元素的 `style` 属性的用法和 DOM 里面的 `style` 不大一样，普通的 HTML 中的：

```
<h1 style='font-size: 12px; color: red;'>React.js 小书</h1>
```

在 React.js 中你需要把 CSS 属性变成一个对象再传给元素：

```
<h1 style={{fontSize: '12px', color: 'red'}}>React.js 小书</h1>
```

`style` 接受一个对象，这个对象里面是这个元素的 CSS 属性键值对，原来 CSS 属性中带 `-` 的元素都必须要去掉 `-` 换成驼峰命名，如 `font-size` 换成 `fontSize`，`text-align` 换成 `textAlign`。

用对象作为 `style` 方便我们动态设置元素的样式。我们可以用 `props` 或者 `state` 中的数据生成样式对象再传给元素，然后用 `setState` 就可以修改样式，非常灵活：

```
<h1 style={{fontSize: '12px', color: this.state.color}}>React.js 小书</h1>
```

只要简单地 `setState({color: 'blue'})` 就可以修改元素的颜色成蓝色。

## 课后练习

\*[覆盖默认样式](#)

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：PropTypes 和组件参数验证](#)

[上一节：props.children 和容器类组件](#)

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

0 条评论，0 人参与。

★ 1



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

还没有评论

友言?