在 React 组件类中如何保证 `this` 为正确的上下文?

作者: [Vanessa](#)

原文链接: <https://ld246.com/article/1552557593163>

来源网站: [链滴](#)

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

回答

在 JavaScript 类中，默认情况下方法并没有被绑定。这意味着他们的 `this` 上下文可以被改变（例如事件处理方法中 `this` 为正在侦听事件的元素）但不排除为组件实例。要解决此问题，可以使用 `Function.prototype.bind()` 强制将组件实例作为 `this` 上下文。

```
constructor(props) {
  super(props);
  this.handleClick = this.handleClick.bind(this);
}
handleClick() {
  // Perform some logic
}
```

- `bind` 不仅啰嗦而且需要定义一个 `constructor`，因此我们可以使用新的公共类字段语法：

```
handleClick = () => {
  console.log('this is:', this);
}
```

```
render() {
  return (
    <button onClick={this.handleClick}>
      Click me
    </button>
  );
}
```

- 你也可以使用内联箭头函数，因为 `this`（组件实例）是受到保护的：

```
<button onClick={e => this.handleClick(e)}>Click me</button>
```

加分回答

- 需要注意的是，使用内联箭头函数会产生额外的再次渲染，因为在渲染时会创建一个新的函数引用这样新的引用就会传递给子组件并且中断 `shouldComponentUpdate` / `PureComponent` 的相等检来阻止不必要的重新渲染。在性能很重要的情况下，最好在构造函数中使用 `bind` 或优先使用公共类段语法，因为他们的函数引用是保持不变的。

返回总目录

[30 秒面试系列一](#)