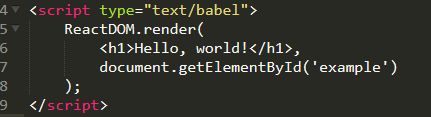
1. **Date.toLocaleTimeString()**

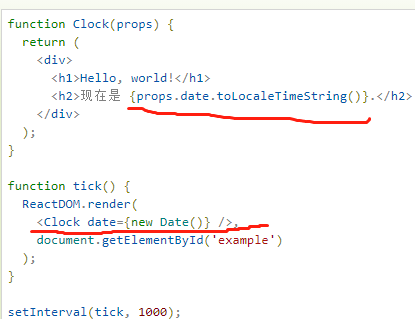
**根据本地时间把 Date 对象的时间部分转换为字符串：**

**二、**



**React.dom.render将dom元素渲染到页面中**

1. **react给组件标签传参数**



**标签名<Clock date={new Date()} />中，标签名必须大写，date绑定的是要穿的参数，render中接受用props，他与函数function Clock（props）{ }的函数名对应**

1. **React.Component 的 ES6 类**

**class Clao(组件名) extends React.Component{**

**render(){**

**return（**

**内容{拿取传输的数据this.props.数据键名}**

**）**

**}**

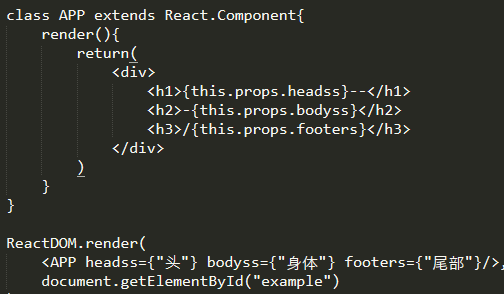
**}**

**Class写法中的props要用this.props来获取传输的值**





1. **class多层 props**

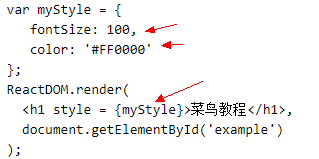


1. **react JSX**

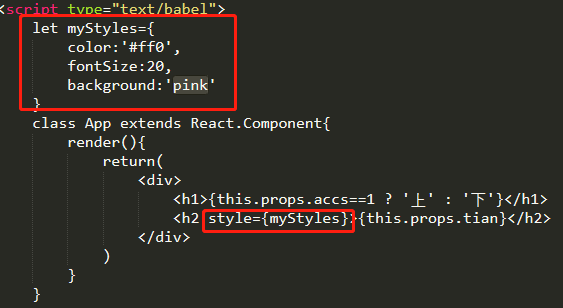
**JSX 就是用来声明 React 当中的元素。**

**标签中可以用data-myattribute添加属性**

1. **camelCase 语法会在指定数字后面加px，**



**箭头1处为逗号  二处不能加； 三处为{}**



**颜色属性要加引号否则报错**

**注释{/\*注释...\*/}**

**八、**super(*props*)

**包含App组件中所有的值**

1. **props**

**state 和 props 主要的区别在于 props 是不可变的，而 state 可以根据与用户交互来改变。这就是为什么有些容器组件需要定义 state 来更新和修改数据。 而子组件只能通过 props 来传递数据。**

**没有 state 的组件叫无状态组件（stateless component），设置了 state 的叫做有状态组件（stateful component）**

**App.defaultProps={times:”阿萨达多”} defaultProps可以能够为组件添加要传输的值**





1. **组件之间相互传值**



**App为父组件，通过props传递数值给Qpp和Www**

**十一、**propTypes

**Props 验证使用 propTypesReact.PropTypes 提供很多验证器 (validator) 来验证传入数据是否有效。当向 props 传入无效数据时，JavaScript 控制台会抛出警告。**



**十二、react事件绑定**

1. **：**



**This.brnc=this.brnc.bind(this)必须添加，否则事件不执行**



**将事件用=（）= >来书写，是实验性的属性初始化器语法 函数内传传参数会出现报错 如：this.brn(“阿萨达”)，将报错**

1. **以在回调函数中使用 箭头函数：**

**函数写在与render平级的地方**

**改变state中的值用：**

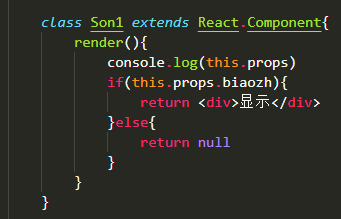
**This.setState(state=>({**

**Time:!this.state.time**

**}))**

**阻止渲染：使组件不显示用return null**

**（）=>this.brnc(“asd1”)可以穿参数 必须加（）**

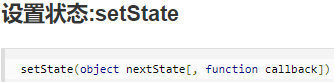




**十三、循环**



**十四、react API**





**十五、React.createElement()**



**十六、state**

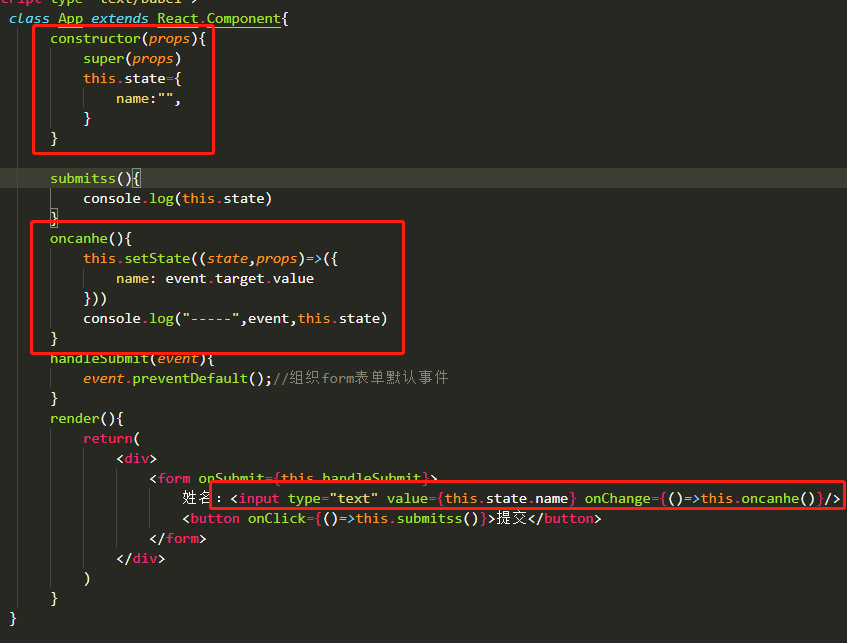
**e.preventDefault()取消默认事件**



**This.setState中可以获取到state和props**

**十七：表单提交**

**1.获取input的输入值**



**①Input中的value绑定this.state中对应的值**

**②设置change事件，在事件中用setState实时获取value值，event.Target可以实时打印出input的输入值**

**③this.state中的值**