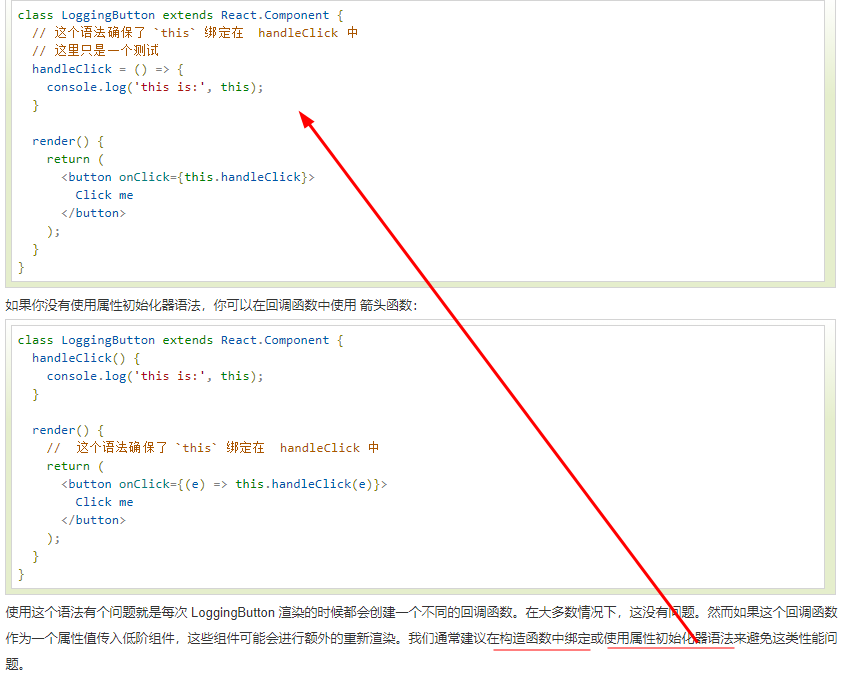
如果你没有在方法后面添加 () ，例如 onClick={this.handleClick}，你应该为这个方法绑定 this。



bind(this)





箭头函数（定义时和调用时）



调用的时候bind可以传递参数

补充：调用时的箭头函数也可以传递参数，不过只能传一个,第二个是事件对象

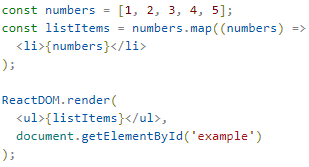


构造函数中：this.state

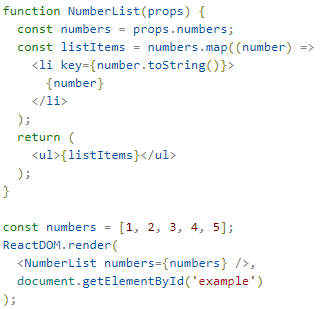
两种方式传递参数



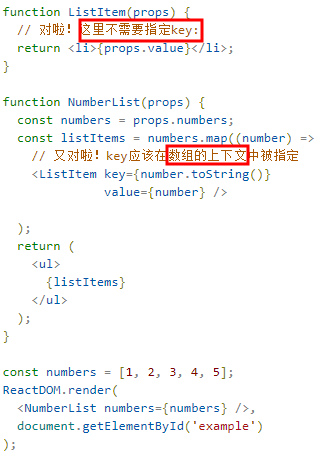
一旦showWarning变化，WarningBanner就会重新渲染



变量存放着数组，数组存放的元素为dom元素



动态渲染列表，函数的参数props是在调用的时候传入的

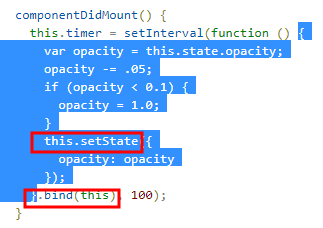


调用的时候才需要key



数组元素中使用的 key 在其兄弟之间应该是独一无二的。然而，它们不需要是全局唯一的。当我们生成两个不同的数组时，我们可以使用相同的键。







类组件，在构造器外边可以直接使用this.props

this.setState({})





在render函数中，表中里可以用this.props

子组件触发父组件方法



select上的值需要通过方法来设置

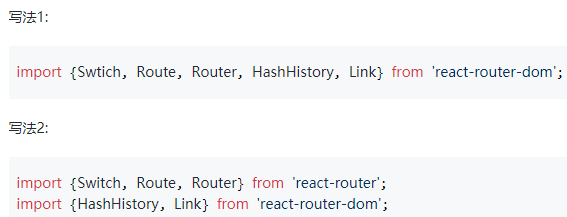


this.setState有变量时

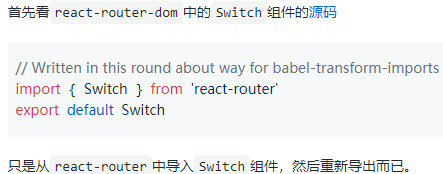


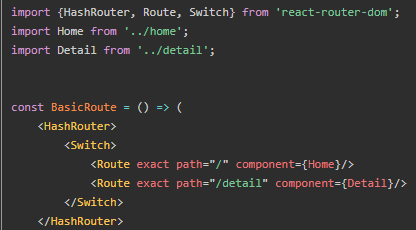
通过refs获取元素

调用的时候bind(this)可以传递多个参数



react-router-dom是web版本的







调用时的箭头函数

调用push函数

history



Route中路径定义变量（动态路由）

动态参数放在props中



history.push()也可以传递参数

this.props.history.replace('/detail');

this.props.history.goBack();

[IndexRoute](http://react-guide.github.io/react-router-cn/docs/guides/basics/docs/API.md#indexroute) 来设置一个默认页面。



import { IndexRoute } from 'react-router'



可以不用这样写/inbox/messages/:id，而是这样写/messages/:id

但是，/inbox/messages/5 都会看到一个错误页面。

（嵌套路由的地址拼接，要拼接那么子路由就不要有跟路径/），但是

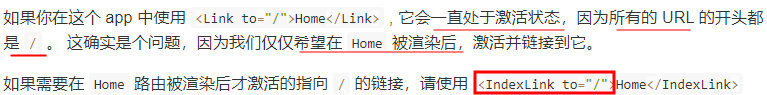
可以用重定向解决



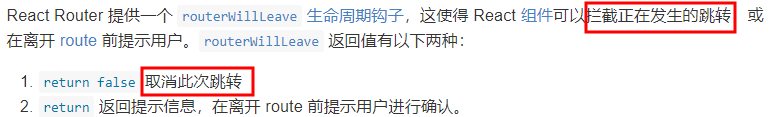
**提醒**：绝对路径可能在[动态路由](http://react-guide.github.io/react-router-cn/docs/guides/basics/docs/guides/advanced/DynamicRouting.md)中无法使用。

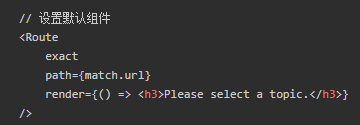
<Router history={browserHistory} routes={routes} />

创建一个像example.com/some/path这样真实的 URL 。



Link一直处于激活状态

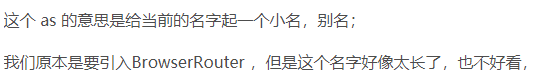




Route中嵌套html

有<Switch>标签，则其中的<Route>在路径相同的情况下，只匹配第一个，这个可以避免重复匹配；

<https://www.cnblogs.com/ipoodle/p/9690305.html>



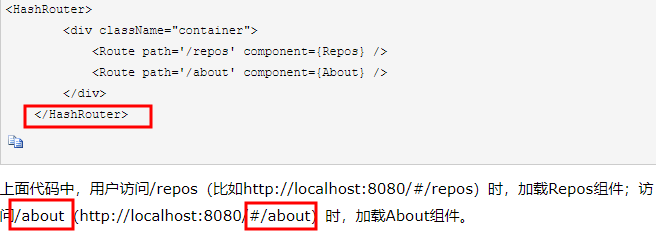
老浏览器提供的 hash模式， 我们称之为： HashRouter

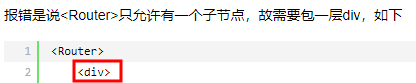
H5提供的的 history 模式，我们称之为 BrowserRouter

注意： H5模式的路由需要后端支持

NavLink( 路由激活时可以用activeClassName 来添加active类名 )

<NavLink activeClassName="active" className="nav-link" to = "/home">Home</NavLink>





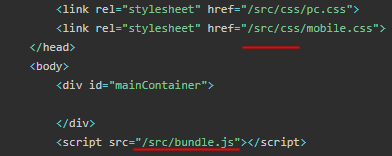
Router只能有一个子节点



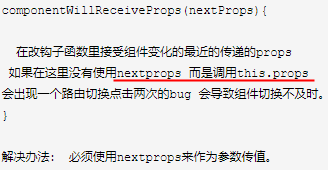


BrowserRouter版本比Router新（变长了）

在当前目录下的文件路径不再使用./, 而是直接用/



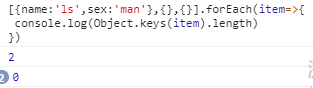
css的引入



钩子函数参数传递props

Jsonp的封装：<https://www.cnblogs.com/helloNico/p/10598667.html>

Jsonp它只支持GET请求而不支持POST等其它类型的HTTP请求



判断对象是否有属性



实战链接：<https://github.com/liupeijun/react-router-v4-maizuo.git>

url地址：<https://github.com/liupeijun/react-router-v4-maizuo>

# 项目名称：[react-router-v4-maizuo](https://github.com/liupeijun/react-router-v4-maizuo)

二：

<https://github.com/deot/react-examples.git>

<https://github.com/deot/react-examples>

[react-examples](https://github.com/deot/react-examples)

三：

<https://github.com/hfpp2012/react-redux-crud.git>

<https://github.com/hfpp2012/react-redux-crud>

[react-redux-crud](https://github.com/hfpp2012/react-redux-crud)

四：

<https://github.com/xbmchina/react-project-news.git>

<https://github.com/xbmchina/react-project-news>

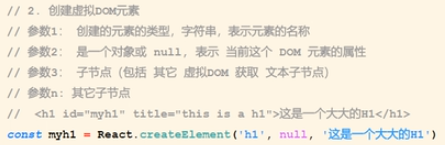
[react-project-news](https://github.com/xbmchina/react-project-news)



把状态交给provider,router也可以有其他非route元素

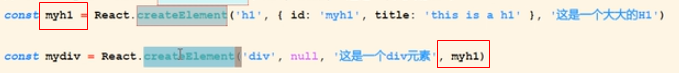
**setState()是异步的，**所以执行this.setState()后立即获取state的值是不变的。





createElement



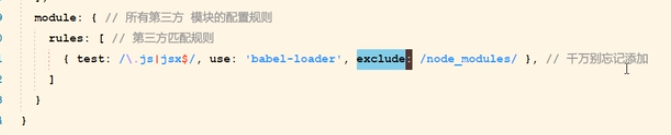


给myh1添加子节点，创建节点，创建元素



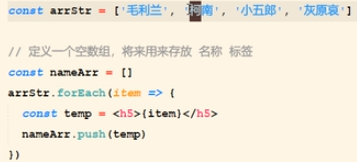


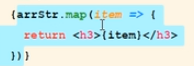
jsx的配置



exclude node\_modules

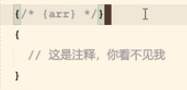








循环数组forEach map





因为跟js有冲突

const mySet = new Set([1, 2, 3, 4, 4])

[...mySet] // [1, 2, 3, 4]//set对象转为数组

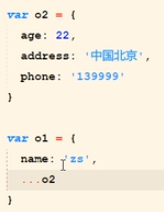
<https://www.bilibili.com/video/av37668737/?p=20>



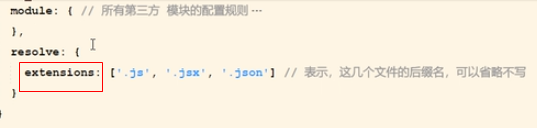
不渲染的时候可以返回null







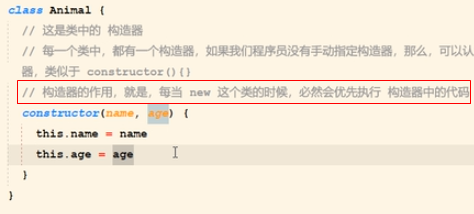
解构赋值



省略后缀名的配置



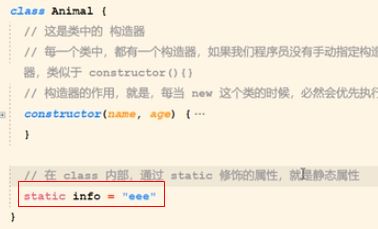
@路径别名



在实例化的时候，构造器优先被执行



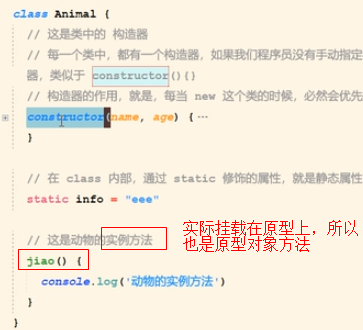
函数的静态属性

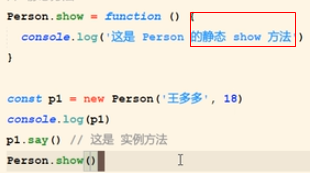


静态属性

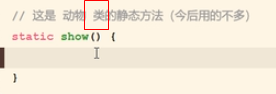


prototype是实例方法





静态方法挂载在构造器上



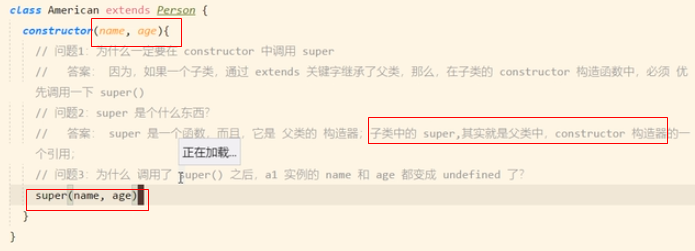
class其实内部的原理还是function,静态的挂载到构造器

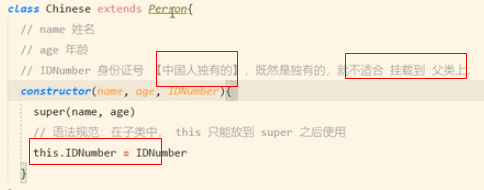


类的继承

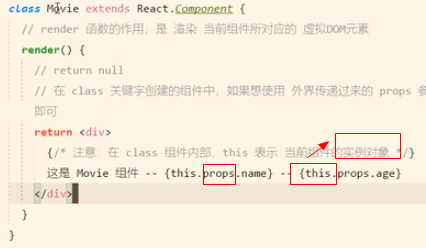


方法也会被继承

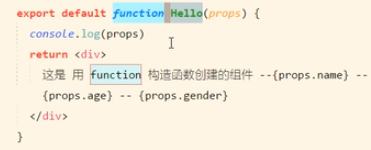






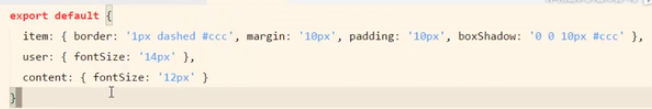


实例对象的属性props



有无状态、有无生命周期函数

但无状态组件的运行效率会高一些，它没有生命周期函数







loader从右往左调用，所以注意一下顺序







模块化

只有id和类名可以模块化



模块化样式配置



不被模块化的样式





图片的loader



引入bootstrap样式



sass-loader node-sass



scss-loader

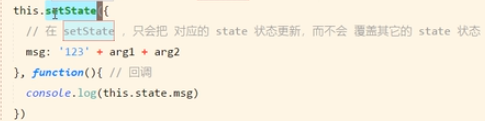


模块化scss

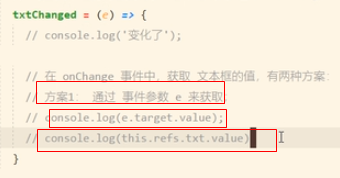


没有模块化，所以不用变量接收





setState是异步的,用回调取到最新的值

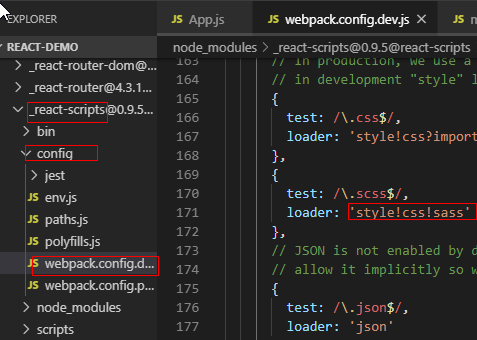


双向绑定

this.refs

<Route path="/about" component={Mine} />

component的c是小写的呐草，搞死人



<https://www.bilibili.com/video/av57303253/?p=4>

react-native