在react中，要将react组件连接到redux中，通常会这样包装组件

class Home extends Component {

}

function select(state) {

return {

logBox:state.logBox

}

}

export default connect(select)(Home)

但是当搭配react-router的时候，在进行路由跳转的时候，组件不会重新render。

**react-redux的connect方法的说明：**

### connect([mapStateToProps], [mapDispatchToProps], [mergeProps],[options])

mapStateToProps(state, [ownProps])： 如果定义该参数，组件将会监听 Redux store 的变化。

mapDispatchToProps(dispatch, [ownProps])： 如果传递的是一个函数，该函数将接收一个 dispatch 函数，然后由你来决定如何返回一个对象mergeProps(stateProps, dispatchProps, ownProps)：如果指定了这个参数，mapStateToProps() 与 mapDispatchToProps() 的执行结果和组件自身的 props 将传入到这个回调函数中。该回调函数返回的对象将作为 props 传递到被包装的组件中。

options：如果指定这个参数，可以定制 connect 的行为。

* [pure = true] (Boolean): 如果为 true，connector 将执行 shouldComponentUpdate 并且浅对比 mergeProps 的结果，避免不必要的更新默认值为 true。
* [withRef = false] (Boolean): 如果为 true，connector 会保存一个对被包装组件实例的引用，该引用通过 getWrappedInstance() 方法获得。默认值为 false

options中pure属性默认为true,估计是因为浅对比的原因，没有获取到路由的变化，因此可以将pure设置为false

class Home extends Component {

}

function select(state) {

return {

logBox:state.logBox

}

}

export default connect(select,undefined,undefined,{pure:false})(Home)

这样路由变化的时候，该组件就能监听并且重新render了。