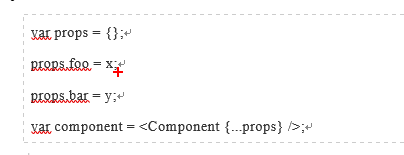
四 视图相关概念

Props 对外的

State 对内的

**4.2.1 jsx语法详解**

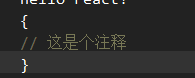
**4.2.4 属性扩散**



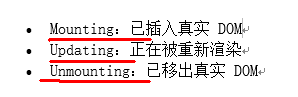
注释 两边要再加一个大括号

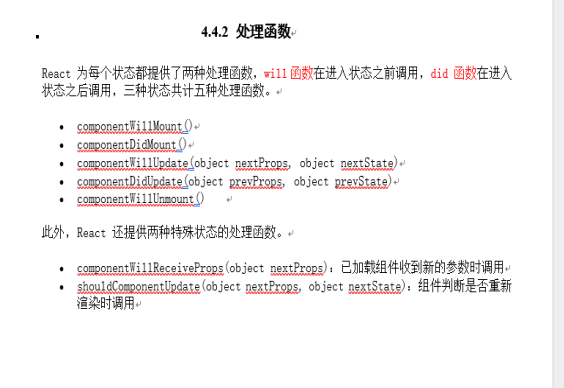
clipboard.png

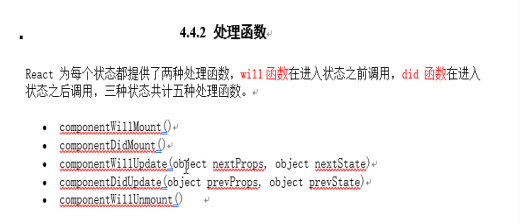
第二种 不能写一行,要不都被注释了



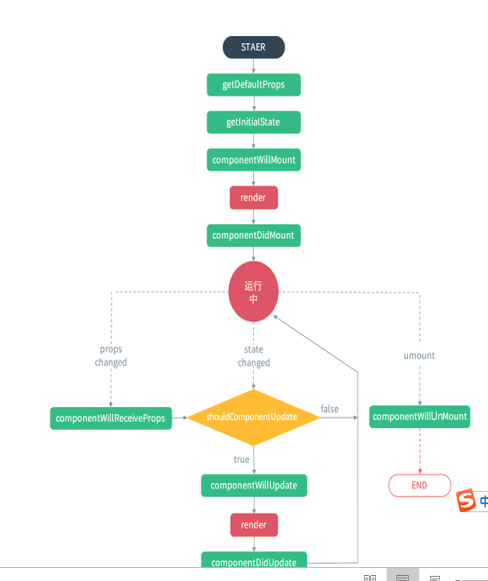
**组件的生命周期**







最重要的 .组件中所有的方法



初始化props

初始化 state

componentWillMount 渲染之前调用的方法.

render ==>渲染用的 把虚拟dom树上内容往真是dom节点上写入,进行渲染

componentDidMount 渲染之后调用的方法 说明界面上就有东西了,就显示了.

底下就不走了,除非是当有交互的时候

当你有交互的时候,开始往下走.

如果更改statechanged这个值

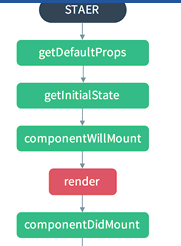
shouldComponentUpdate 是个判断,如果是false ,就会返回不动的状态.

如果是true 就会继续向下走

componentWillUpdate

render 又调用了一遍 . 重新进行渲染,和上面的render 是一个.

componentDidUpdate



这部分只会运行一次

底下部分是会运行多次

如果更改的是propschanged 这个值 ,调用componentWillReceiveProps

如果当前组件被干掉的时候 . 走umount => componentWillUnMount也只会走一次.

ES5的组件的写法 是分2部分的 .

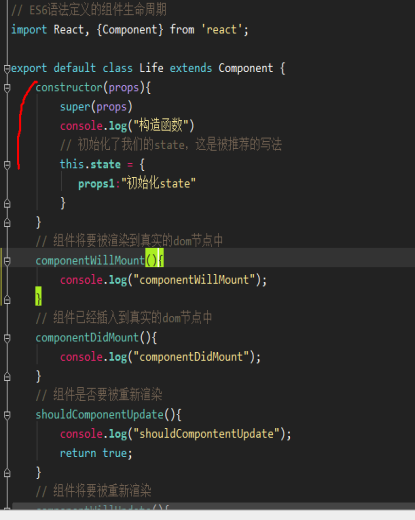








**ES6**

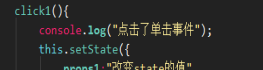


**constructor 初始化 .**

clipboard.png

**这是固定写法,写constructor 必须写supper 属于也初始化了component中的chonstructor**

方法1

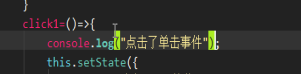


**如果想调用这个click1 ,必须以下这么写**

clipboard.png

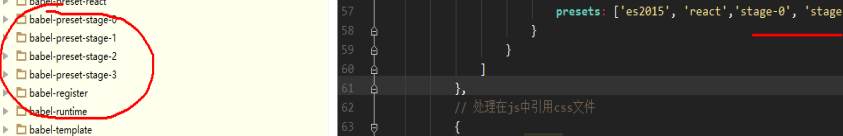
**方法二**

**如果上边是箭头函数,后边就不用bind**



clipboard.png

**如果用箭头函数报错了,就是忘记安stage-0这几个了**



注意



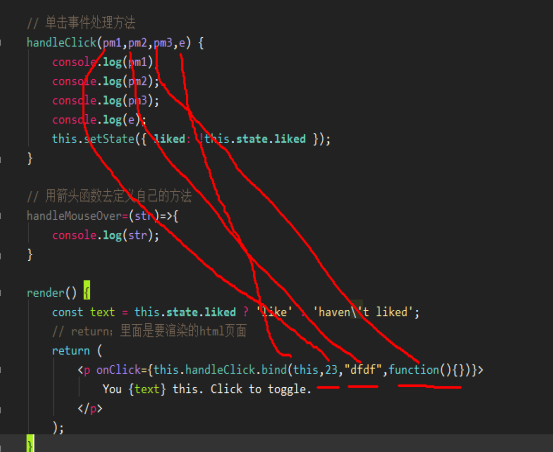
重新渲染一定要给返回值true

[www.tuicool.com](http://www.tuicool.com)

es5 和es 6 的react的写法

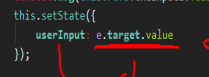
参数传递

顺序注意 ,bind this ,是在参数中最后一个接收,就是e



**方法二 refs属性 获取dam 元素**

**render中只能有一个根,也就是一个div.其他的写在这个div中**



**e.target.value 键盘上输入的值**

**注意:**

**render 和render上面是拿不到dom元素节点的**

**下面才可以.**

**组件的dom监听和移除**

**添加事件监听,一定要记得移除**



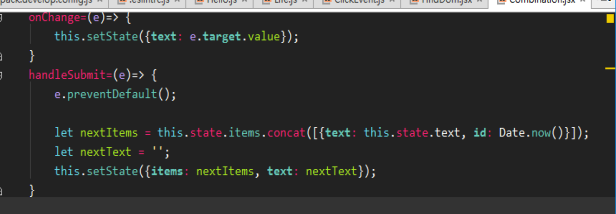
**如果不添加移除,就是bug**

**受控组件**

**表单元素**

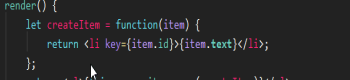
**比如input的value值和state的属性进行绑定了,那么这个inpu就是受控组件.**

**4.13 组合组件**



input 清空输入框,让输入框中的 值,显示在页面

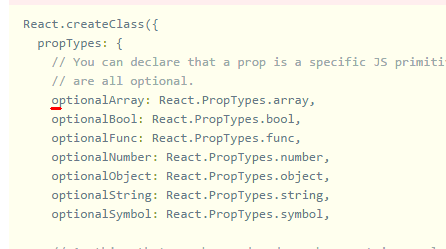
react中类似于列表的. 一定要给一个key值,否则会有警告信息的



4.14 **propsType 属性校验**

**两方传递,a方传递cccc给b,b接收,b接收cccc要确定是否自己接收的就是cccc,所有用propsType进行检测**

**所以===>接收方检测**



**以上是固定写法**

clipboard.png

**这个delete相当于Button的子集,children,如果此位置是个标签,那子集就是个标签**

**Context 解决层层包裹的方式而诞生的**

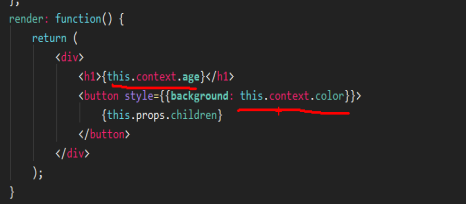
**用法 getChildContext 这种声明方式**



**检验contextTypes 检验 color 是不是字符串类型的**



**中间不需要传递. 声明了.**



**这里就可以直接使用,就是可以拿来用了 .**

**4.16 动画**

**支付宝 ,需要跟支付宝付款,会拿回来很多id .**

**付款框架**

**ping++ 是一个框架 .**

<https://www.pingxx.com/>

注册成开发者

下载 SDK

分为客户端和壹收款 ,功能差不多,名字不一样.

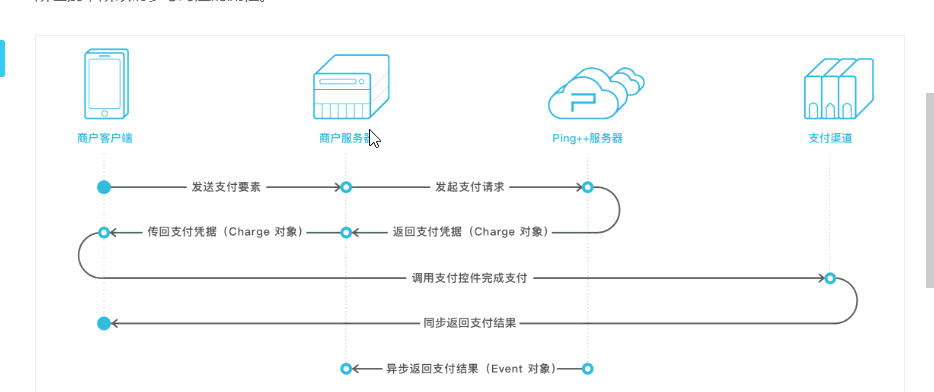
API文档

开发指南:

我们主要看概述和基础

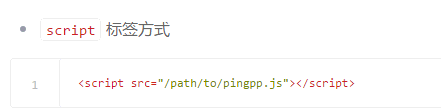
client端接入

如果是server端接入,让服务器端的人处理



自己的手机 => 自己的本地服务器=> ping++的服务器 ==>返回给我们商户服务器一个支付凭据(charge对象) =>本地服务器拿到之后=>返回本地客户端=> 本地客户端将charge对象 提交给支付渠道,比如(微信或者支付宝服务器)==>等待p++最终的返回,

引入js文件



将要用的地方导入ping++



然后导入

