# [React组件之间的传值](https://blog.csdn.net/xsuperman9527/article/details/83145698)

2018年10月18日 14:07:47 [阳光下有彩虹](https://me.csdn.net/xsuperman9527" \t "https://blog.csdn.net/xsuperman9527/article/details/_blank) 阅读数：56

父组件

import React, { Component } from 'react';

import ChildrenOne from './ChildrenOne';

import ChildrenTwo from './ChildrenTwo';

import './father.css';

export default class Father extends Component {

state = {

fatherContent: 'fatherContent', *// 父子*

childrenContent: '' *// 子父,兄弟*

}

handleChange = (text) => { *// 子传父,兄弟之间的传值要先经过父组件再传给子组件*

this.setState({ childrenContent: text });

}

render() {

const {fatherContent, childrenContent} = this.state;

return(

<div className='father'>

<h1>Father</h1>

<div>子组件: {childrenContent}</div>

<ChildrenOne handleChange={this.handleChange} fatherContent={fatherContent} childrenContent={childrenContent} />

<ChildrenTwo handleChange={this.handleChange} fatherContent={fatherContent} childrenContent={childrenContent} />

</div>

)

}

}

子组件1:

import React, { Component } from 'react';

export default class ChildrenOne extends Component {

render() {

*// 点击按钮之后将子组件的值传递父组件,父组件再传给兄弟组件*

const {fatherContent, childrenContent, handleChange} = this.props;

return (

<div className="child">

<h2>ChildrenOne</h2>

<div>父组件: {fatherContent}</div>

<div>兄弟组件: {childrenContent}</div>

<button onClick={() => handleChange('ChildrenOne')}>ChildrenOne</button>

</div>

)

}

}

子组件2和子组件1一致,这里其实可以将两个组件提成一个公共的组件的,但是为了方便我们看传值就先不管了

import React, { Component } from 'react';

export default class ChildrenTwo extends Component {

render() {

*// 点击按钮之后将子组件的值传递父组件,父组件再传给兄弟组件*

const {fatherContent, childrenContent, handleChange} = this.props;

return (

<div className="child">

<h2>ChildrenTwo</h2>

<div>父组件: {fatherContent}</div>

<div>兄弟组件: {childrenContent}</div>

<button onClick={() => handleChange('ChildrenTwo')}>ChildrenTwo</button>

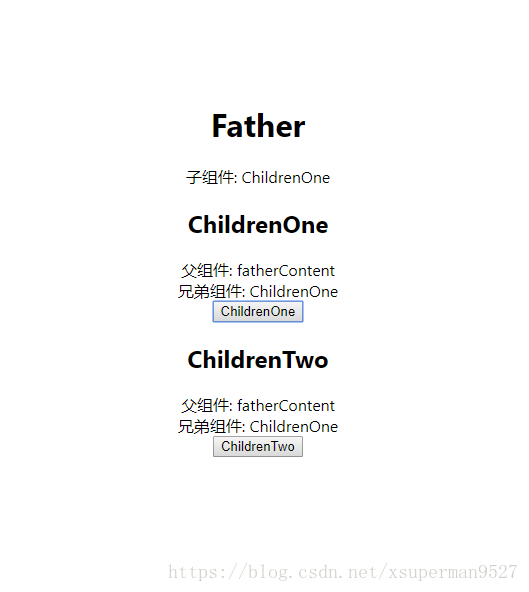
</div>

)

}

}

效果:



点击ChildrenOne之后,子组件1将父组件的值和兄弟组件的值都改变成了''ChildrenOne''。

代码的效果已经出来了,现在我们来分析:

首先说下父子之间的传值:

父传子:

这个主要是通过props来传递的,比较简单,没有什么可以说的。

子传父:

主要是通过回调函数来向父组件传值,从代码中我们可以看到,父组件其实是向子组件传递了一个带参数的函数,我们通过在子组件中调用这个函数就可以达到子向父的传值了。(有没有点感觉像jsonp)

兄弟传值：

主要是通过子传父之后，父再传给子，其实只要理解了子传父，那么兄弟之间的传值就很好理解了。

因为react是单向数据流的关系，一旦形成多层组件嵌套，那么父子，兄弟之间的传值将变得很麻烦，这也是为什么需要redux的原因，后面准备写一个简单的redux的例子。