大地基础

- react简介：
- 使用：
- React目录结构简介：
- super关键字：
  - http://www.phonegap100.com/thread-4911-1-1.html
  - ES6中的super可以用在类的继承中，super关键字，它指代父类的实例（—即父类的this对象—）。子类必须在constructor方法中调用super方法，否则新建实例时会报错。这是因为子类自己没有自己的this对象，而是继承父类的this对象，然后对其进行加工。如果不调用super方法，子类就得不到this对象。
    - 例：
      - class Person{
        - constructor (name){
          - this.name=name;
        - }
      - }
      - class Student extends Person {
        - constructor(name,age){
          - super();//用在构造函数中，必须在使用this之前调用
          - this.age=age;
        - }
      - }
  -
- 组件名称首字母大写、组件 类名称首字母大写
- 所有 的模板需要被一个根节点包含起来。
- {}用来绑定数据　this.state
- 绑定属性：（注意class，和label里面的for，style）
- 引入图片：
  - 引入本地图片：
    - 方法一：es6
      - import **new1Img** from '../assets/images/1.jpg' 引入图片

        - 使用
    - 方法二：AMD规范引入方式
        -
- 引入远程图片：
    -
- 数组操作： 循环数据要加key
- 绑定事件：
- 事件对象：在触发DOM上的某个事件时，会产生一个事件对象event。这个事件对象中包含着所有与事件相关的信息。
- 表单事件：
    - 实现双向数据绑定
        - 1监听表单的改变事件
        - 2在改变事件里获取表单输入的值（事件对象、ref）
        - 3把表单输入的值赋给state里面的username
        - 4点击按钮时获取state里面的username
    - onChange
        - 1通过事件对象获取input的value
        - 2通过ref获取input的value
    - 键盘事件：onKeyUp / onKeyDown / onKeyPress **e.keyCode**
- **React约束性组件和非 约束性组件 （表单组件）**
    - 约束性组件，由react管理它的value。
        - <input type="text" value={this.state.name} onChange={this.handleChange} />
        - handleChange : function(e){
            - this.setState({name : e.target.value });
        - }
        - 上面这个value属性由this.state.name，进而由this.handleChange负责管理的。实际上input的value不是用户输入的内容 。而是onChange事件触发之后，由this.setState决定的。
        - 约束性组件显示的值和用户输入的值虽然在很多时候是相同的，但他们是两码事。约束性组件显示的是this.state.name的值。你可以在handleChange中对用户输入的值做任意的处理，比如错误检验。
    - 非约束性组件，由原生DOM管理它的value
        - <input type="text" defaultValue="a" />

        - 上面这个defaultValue其实就是原生DOM中的value属性。这样写出来的组件，其value值就是用户输入的内容，React完全不管理输入的过程。
    - 对比约束性组件和非约束性组件的输入流程
        - 非约束性组件：用户输入A ——> input中显示A
        - 约束性组件：用户输入A ——> 触发onChange事件——> handleChange中设置state.name = " A "——>渲染input使它的value变成A
    - React把input、textarea和select三个组件做了封装和抽象，他们有统一的value属性和onChange事件。
    - **checkbox改变的不是value，而是checked状态。**
    - 示例：
        - https://www.qdfuns.com/article/40901/34ef7865a8431818697b8dc4956cb062.html
- **react中的组件：解决HTML标签构建应用的不足**
    - 使用组件的好处，把公共的功能单独抽离成一个文件作为一个组件，哪里使用就在哪里引用。
- 父子组件：组件的相互调用中，，我们把调用者称为父组件，被调用者称为子组件
    - 父组件给子组件传值：
        - 1.在调用子组件的时候定义一个属性 <Header msg='首页'></Header>
          2.子组件里面 this.props.msg 说明：父组件不仅可以给子组件传值，还可以给子组件传方法，以及把整个父组件传给子组件，并且可以实现子组件给父组件传值（父组件方法.bind(this,‘子组件的值’）其实就是通过方法调用传参实现）
    - 父组件主动获取子组件的数据
        - 1、调用子组件的时候指定ref的值 <Header ref='header'></Header>
          2、通过this.refs.header 获取整个子组件实例 （要保证DOM组件已经加载完成）
- defaultProps和proptypes_____
-

幕布 - 思维概要整理工具