**React-router**

Router本身只是一个容器，真正的路由要通过Route组件定义

参数：history=”hashHistory / browserHistory” 用来监听浏览器地址栏的变化

hashHistory路由的切换由URL的hash变化决定

React-router提供了一个routerWillLeave生命周期钩子，可以拦截正在发生的跳转，或者在离开route前提示用户。它的返回值有：

return false; 取消此次跳转

return 返回提示信息，在离开route前提示用户进行确认。

React组件中的数据：

React中的数据分为两种：state( 组件的内部状态 )和prop( 组件对外的接口 )

React生命周期函数：

Render 返回的结果用于构造DOM对象

componentWillReceiveProps状态改变检测机制，当父组件再次render的时候子组件会执行该生命周期钩子

shouldComponentUpdate(nextProps, nextState) 返回一个布尔值，告诉react库这个组件在这次更新过程中是否要继续， 如果返回结果为true，继续更新过程，接下来调用render函数；返回结果为false，立刻停止更新过程，也就不会引发后续的渲染了。

componentWillUnmount() 当react组件要从dom树上删除掉之前会调用，适合做一些清理性的工作。

Webpack:

代码分割：使用require.ensure()

React项目中的UI框架选择：

1. bootstrap：

本身包很大，它是在入口文件app.js中提前把css，js文件引入进去，即使在项目中有的css,js没有用到。这样的话当打包的时候最后会生成多余的代码，影响加载速度，影响性能。

1. Antd：

antd提供了babel-plugin-import插件,可以实现按需调用需要的css以及js，需要用 哪一个组件，import进去就行。

|  |
| --- |
| Babel-plugin-import的使用：   1. yarn add react-app-rewired --dev-save ( 修改create-react-app创建项目进行自定义配置 ) 2. 修改package.json 中的启动配置：      1. Yarn add babel-plugin-import --dev-save (实现按需加载组件代码和样式的babel插件) 2. 在项目根目录创建config-overrides.js文件，并修改默认配置： 3. 重启项目：yarn start |