vds5-知识-react.boots.V1.01

（2016/8/22）

目录

[vds5-知识-react.boots.V1.01 1](#_Toc459652424)

[1 安装方法 3](#_Toc459652425)

[1.1 Node & Npm环境 3](#_Toc459652426)

[1.2 预编译模式安装 3](#_Toc459652427)

[2 react-bootstrap常用控件 5](#_Toc459652428)

[2.1 ButtonToolbar & Button 5](#_Toc459652429)

[2.2 FormGroup & FormControl & ControlLabel & HelpBlock 6](#_Toc459652430)

[2.3 Checkbox & Radio 7](#_Toc459652431)

[3 编译代码 8](#_Toc459652432)

# 安装方法

## Node & Npm环境

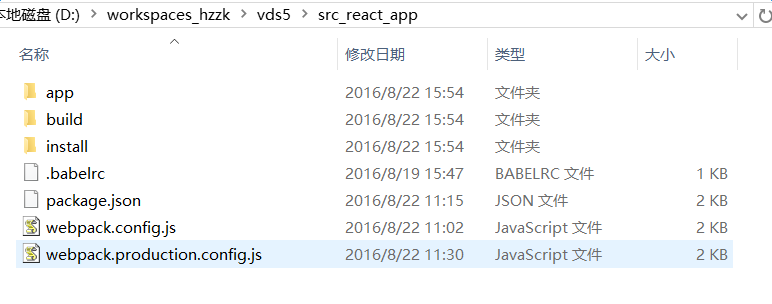
Node安装包路径: \\vds5s1\_src\install\node-v4.4.5-x64.msi, NPM（node package manager），称为node包管理器，Node安装包自带NPM工具。

本实例使用Node版本为v4.4.5,NPM版本为2.15.5。

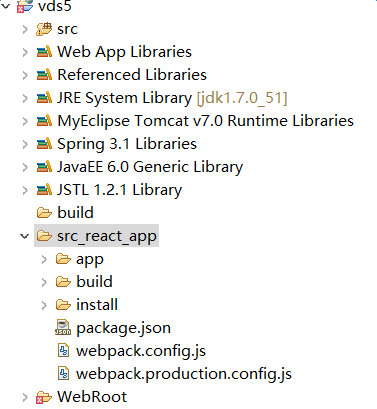
参考资料<http://www.tuicool.com/articles/VB7nYn>

## 预编译模式安装

第一步、解压src\_react\_app.rar 如下：



第二步、将这些文件copy到eclipse或tomcat下，如下图：



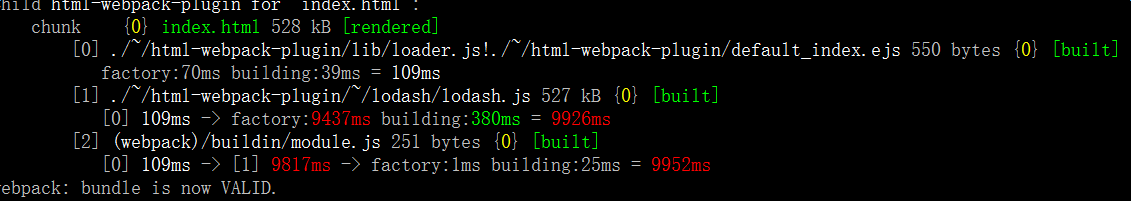
第三步、在cmd控制台进入src\_react\_app目录，并执行npm install命令安装项目依赖包



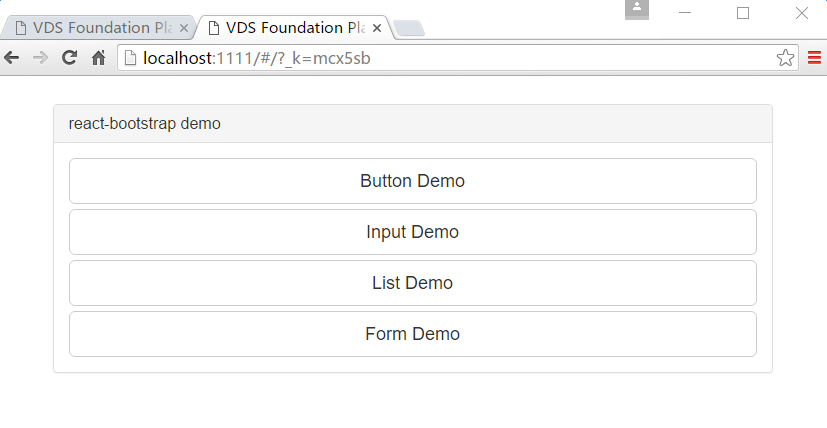
第四步、等待依赖包安装完全后执行npm run dev运行项目



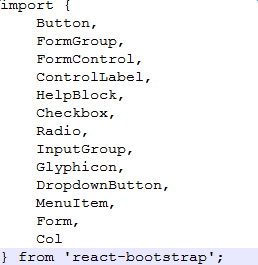
第五步、看到如下图片出现的信息则node服务端开启成功，



浏览器访问<http://localhost:1111/> ，将会看到下面图片内容，表示运行成功。



# react-bootstrap常用控件

使用到的控件的需要导入：（不import就用会报错）

参考资料：<https://react-bootstrap.github.io/components.html>

## ButtonToolbar & Button

一般ButtonToolbar和Button结合使用

例子：

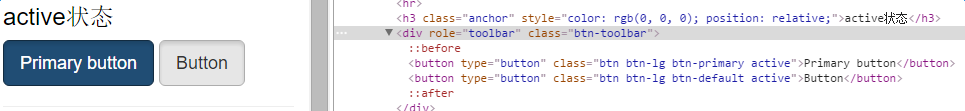
<ButtonToolbar>

<Button bsStyle="primary" bsSize="large" active>Primary button</Button>

<Button bsSize="large" active>Button</Button>

</ButtonToolbar>

渲染效果：



<Button />主要属性：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| active | boolean | False | 是否激活状态 |
| block | boolean | False | 是否为block块 |
| bsSize | "lg", "large", "sm","small", "xs", "xsmall" |  | 按钮尺寸 |
| bsStyle | "success", "warning","danger",  "info", "default","primary", "link" | 'default' | 按钮样式类型 |
| disabled | boolean | False | 是否禁用 |
| href | String |  | 链接 |
| onClick | Function |  | 点击事件 |
| type | 'button', 'reset','submit' |  | 按钮事件类型 |

## FormGroup & FormControl & ControlLabel & HelpBlock

FormGrouph可以和FormControl, ControlLabel, HelpBlock结合使用

例子：

<FormGroup controlId="formValidationSuccess1" validationState="success">

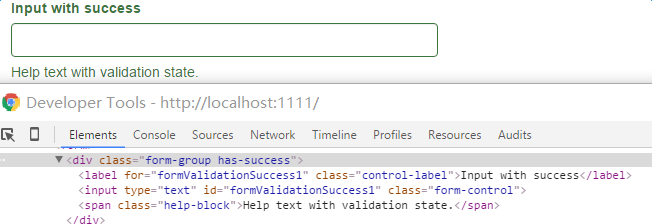
<ControlLabel>Input with success</ControlLabel>

<FormControl type="text" />

<HelpBlock>Help text with validation state.</HelpBlock>

</FormGroup>

渲染效果：



< FormGroup />主要参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| controlId | string |  | 设置里面<FormGroup>  标签的id |
| validationState | 'success','warning', 'error' |  | 验证的状态 |
| bsSize | "lg", "large", "sm","small", "xs", "xsmall" |  | 按钮尺寸 |

< FormControl/>主要参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| id | string |  | 如果这个属性为空，则取<FormGroup />的controlId属性 |
| placeholder | String |  | 输入框默认提示值 |
| componentClass | elementType | 'input' | 默认渲染成input输入框 |
| type | 'text','email', 'password ' |  | Input输入框类型 |

< ControlLabel />与< HelpBlock />主要参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| bsClass | string | < ControlLabel /> : 'control-label'  < HelpBlock /> : 'help-block' | css class |

## Checkbox & Radio

Checkbox和Radio可以和FormGroup组合使用

例子：

<FormGroup validationState="success">

<Checkbox inline>

Checkbox

</Checkbox>

{' '}

<Checkbox inline>

with

</Checkbox>

{' '}

<Checkbox inline>

success

</Checkbox>

</FormGroup>

渲染效果：

<Checkbox />主要属性：

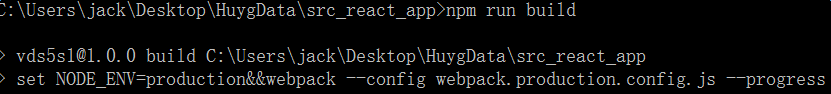
|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| bsClass | string | 'checkbox' | css class |
| disabled | boolean | false | 输入框默认提示值 |
| inline | boolean | false | 默认渲染成input输入框 |
| validationState | 'success','warning', 'error' |  | 验证的状态（inline 属性设置此属性无效） |

< Radio />主要属性：

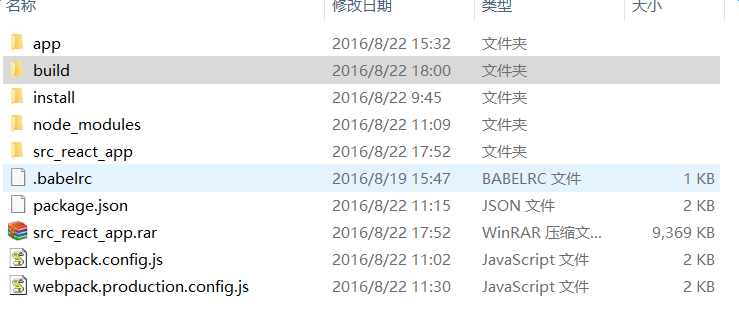
|  |  |  |  |
| --- | --- | --- | --- |
| 属性名 | 类型 | 默认值 | 解释 |
| bsClass | string | 'radio' | css class |
| disabled | boolean | false | 输入框默认提示值 |
| inline | boolean | false | 默认渲染成input输入框 |
| validationState | 'success','warning', 'error' |  | 验证的状态（inline 属性设置此属性无效） |

# 编译代码

cmd控制台执行npm run build：



执行成功后将生成build文件夹：



js,css及其它资源文件保存在build/assets文件夹

