1. **Vue2.0 + webpack 工程环境从无到有：**

使用**Vue-cli**脚手架（属于Vue全家桶）快速构建项目

【1】首先需要先安装好**node.js**；

【2】安装**webpack** 输入指令 **$ npm install -g webpack**

【3】安装 **$ npm install -g vue-cli**  //全局安装**Vue-cli**

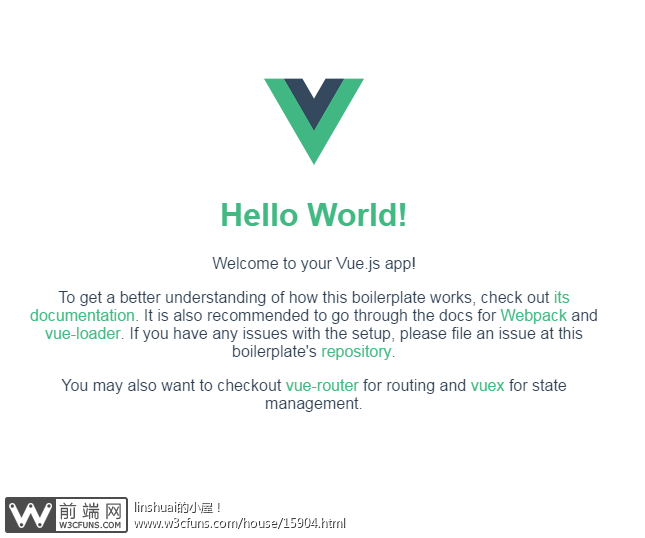
**【4】 $ Vue init webpack projectName** //生成项目名为**projectName** 的新工程

【5】 **$ cd projectName**  //进入工程目录

【6】 **$ npm install** //初始化依赖安装

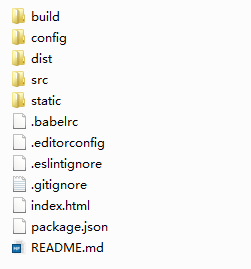
**【7】$ npm install style-loader css-loader url-loader babel-loader sass-loader less-loader file-loader --save-dev** //安装相关加载器用于对应编译

【8】执行 **$ npm run dev** 打开浏览器 <http://localhost:8080> 就可以看到欢迎页面了



【9】 执行 **$ npm run build** 是打包工程，最终上线代码在**dist**目录下；

【10】基本目录结构：主要关注 **index.html** 入口，**src** 开发区，**dist** 发布区，**static** 静态资源



【11】如果出现**index.html**里引用的**css**和**js**的引用路径不对，这时候就需要自己修改一下配置，进入**config / index.js**

原来的配置的引用路径为：

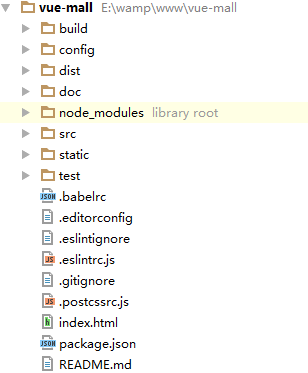


我自己更改为：



这样就能正常访问了 **done!**

1. **Vue-mall 微信商城一级目录：**



**【bulid】**编译发布文件的各种配置文件；

**【config】**配置目录，包括端口号等；

**【dist】**最终发布的代码存放位置，发布区；

**【doc】**项目文档，可在这里编写开发文档；

**【node\_modules】npm**加载的项目依赖模块；

**【src】**这里是我们要开发的目录，基本上要做的事情在这个目录里面，开发区；

**【static】**静态资源目录 如图片、字体等；

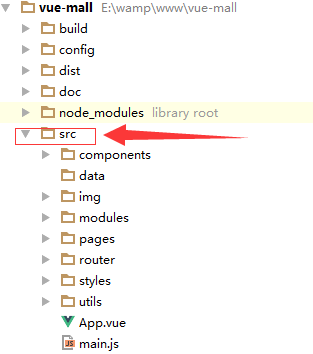
**【test】**初始测试目录，不重要；

**【package.json】**项目配置文件；

**【index.html】**索引文件，入口文件；

**【README.md】**项目的说明文档，**markdown**格式，不重要；

1. **Vue-mall 微信商城src目录：**



**【components】**不含业务功能UI组件放这里，类似弹框，地址联动等组件参考**[YDUI](http://vue.ydui.org/)**；

**【data】**假数据JSON文件；

**【dist】**最终发布的代码存放位置，发布区；

**【img】**项目用到的图片发这里，小于30k的图片会编译为base64存入css文件里面；

**【modules】**带业务数据的模块，通常配合引用【**components】**组件加入数据交互封装成模块，后续可以复用；

**【pages】**项目的页面模版；

**【router】**配置页面路由的地方；

**【styles】**写样式的地方，主要是**chokui.less** 和 **style.less**；

**【utils】**工具类；

**【App.vue】**项目入口文件；

**【main.js】**项目的核心文件，全局的配置都在这个文件里面配置，包括引入模块，组件注册；

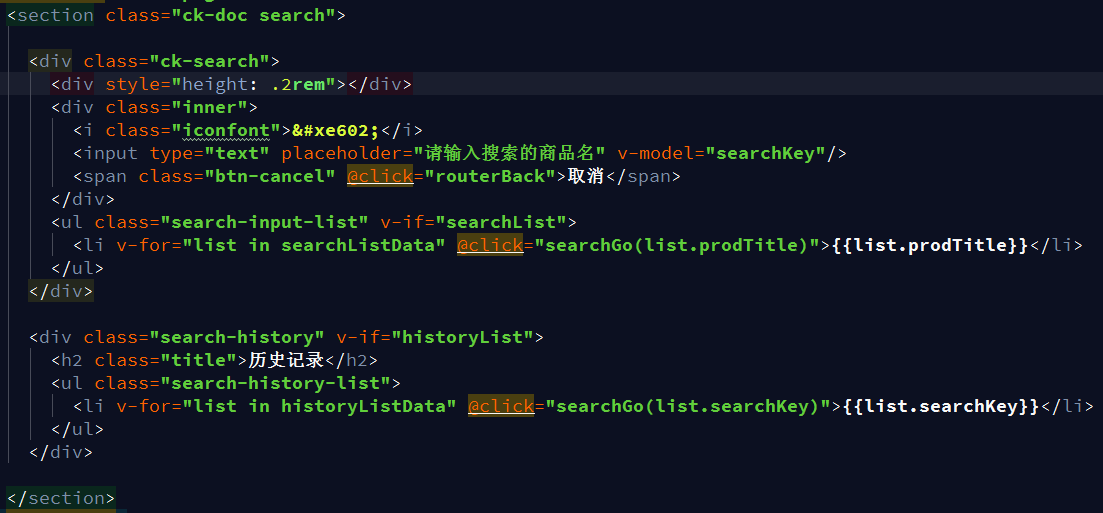
1. **Vue2.0生命周期和钩子函数的一些理解：**

**<https://segmentfault.com/a/1190000008010666>**

**[https://cn.vuejs.org/v2/api/#实例方法-生命周期](https://cn.vuejs.org/v2/api/" \l "实例方法-生命周期)**



**5. Vue2.0常用一些方法（最好看项目代码更为清晰）：**



**【v-html】类似jQuery 的 .html(str) 方法**

**<span v-html="message"></span>**

**【v-bind】绑定属性**

**<span v-bind:id="message"></span>**

**简写 =>**

**<span :id="message"></span>**

**【v-on】事件绑定，类似jQuery 的 .on() 方法**

**<div v-on:click="doSomething"></div>**

**简写 =>**

**<div @click="doSomething"></div>**

**【v-if】条件判断**

**【v-for】 遍历数组**

**【v-text】**

**<span>{{message}}</span>**

**等价 =>**

**<span v-text=”message”></span>**

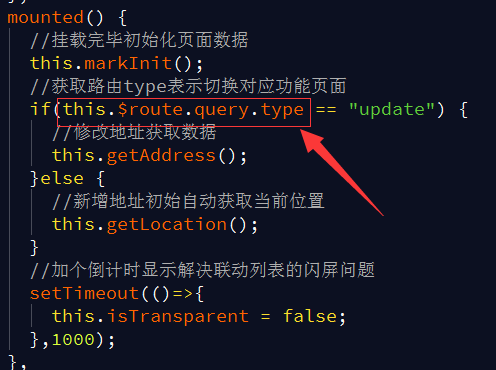
属于**vue**指令方法，下面地址可以了解具体用法：

**[https://cn.vuejs.org/v2/api/#指令](https://cn.vuejs.org/v2/api/" \l "指令)**

**【$router】**看代码很清楚怎么用了



**从** **$router 中取值**

  
属于 **vue-router** 模块提供的路由方法，下面地址可以了解具体用法：

**<https://segmentfault.com/q/1010000007154489>  
<http://router.vuejs.org/zh-cn/index.html>**

**【$http】**异步请求数据

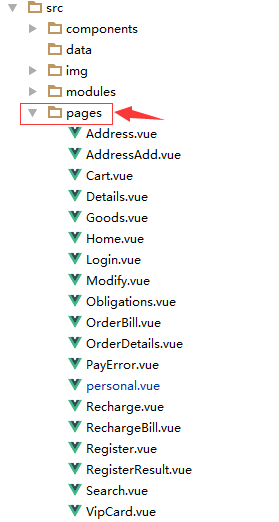


属于 **vue-resource** 模块提供的请求方法，下面地址可以了解具体用法：

**<http://www.doc00.com/doc/1001004eg>**

**<https://github.com/pagekit/vue-resource/blob/master/docs/http.md>**

**6. Vue-mall 微信商城src/page目录：**



**【Address.vue】**个人中心（地址列表）模块 **【AddressAdd.vue】**地址修改/新增模块

**【Cart.vue】**购物车模块

**【Details.vue】**商品详情模块

**【Goods.vue】**商品列表模块

**【Home.vue】**商城首页模块

**【Login.vue】**登录模块

**【Obligations.vue】**个人中心（待付款列表/全部订单）模块

**【OrderBill.vue】**订单结果详情模块

**【OrderDetails.vue】**订单支付详情（立即支付/购物车支付进入的页面）模块

**【PayError.vue】**订单支付失败模块

**【Personal.vue】**个人中心首页模块 **【Modify.vue】**个人中心（个人资料修改）模块

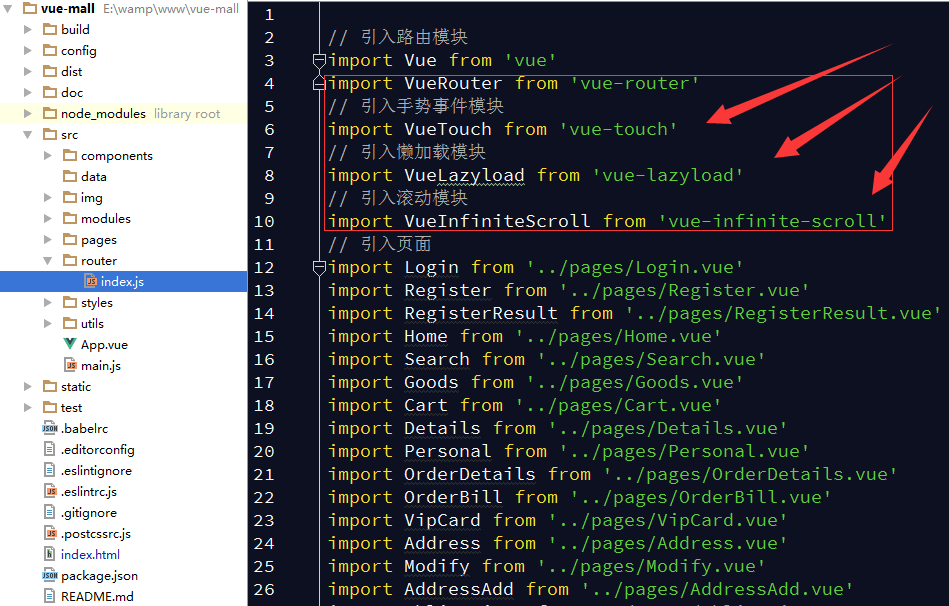
**【Recharge.vue】**会员充值列表模块 **【RechargeBill.vue】**会员充值结果模块

**【Register.vue】**会员注册模块 **【RegisterResult.vue】**会员注册结果模块

**【PayError.vue】**订单支付失败模块

**【Search.vue】**搜索商品模块 **【VipCard.vue】**会员卡模块

1. **Vue-mall 微信商城用到的 npm模块：**



模块 **vue-touch** 提供的手势事件方法，下面地址可以了解具体用法：

**<https://github.com/vuejs/vue-touch>**

模块 **vue-lazyload** 提供的懒加载图片的方法，下面地址可以了解具体用法：

**<https://github.com/hilongjw/vue-lazyload?utm_source=tuicool&utm_medium=referral>**

模块 **vue-infinite-scroll** 提供的滚动加载的方法，下面地址可以了解具体用法：

**<https://github.com/ElemeFE/vue-infinite-scroll>**