Vue-router的配置、使用

前提：vue项目内要有vue-router插件

1. npm I vue-router –save
2. 使用vue-cli创建项目时配置

在使用vue-cli2创建项目的时候配置了路由，发现项目的src目录中多了router目录，router目录就是专门处理前端路由的目录，默认有一个index.js文件。



本demo会删除自动创建的router目录，自己写一遍

其实之前学node中已经接触到了后端路由，其实 前后端路由都是 。请求的url与对应的业务逻辑之间的映射关系。

在学习node中的express-generator时，自动生成的express项目也会自动创建一个管理路由的目录，只不过名称为routes（vue-cli自动生成router），而且也要引入express对象，实例化使用。配置完路由后，又会去入口文件使用该路由。

**配置过程:**

1. 创建router目录（存放各个路由配置）
2. 创建index.js（默认路由）
3. 引入插件，引入vue.js（需要使用Vue.use下载插件）
4. 实例化插件类并在参数对象中配置routes、mode:”history”
5. 默认导出实例化插件对象

之前在学习vue基础的时候也遇到过该问题，**引入vue相关的文件和插件**，使用ES6模块规范才有效，使用**CommonJS规范会报错**。

在配置路由前，先创建好对应的组件，因为要配置映射关系。



实例化vue-roter类，在参数对象中配置routes，routes为Array,元素为obj，obj中配置请求的url与抽离组件的映射关系

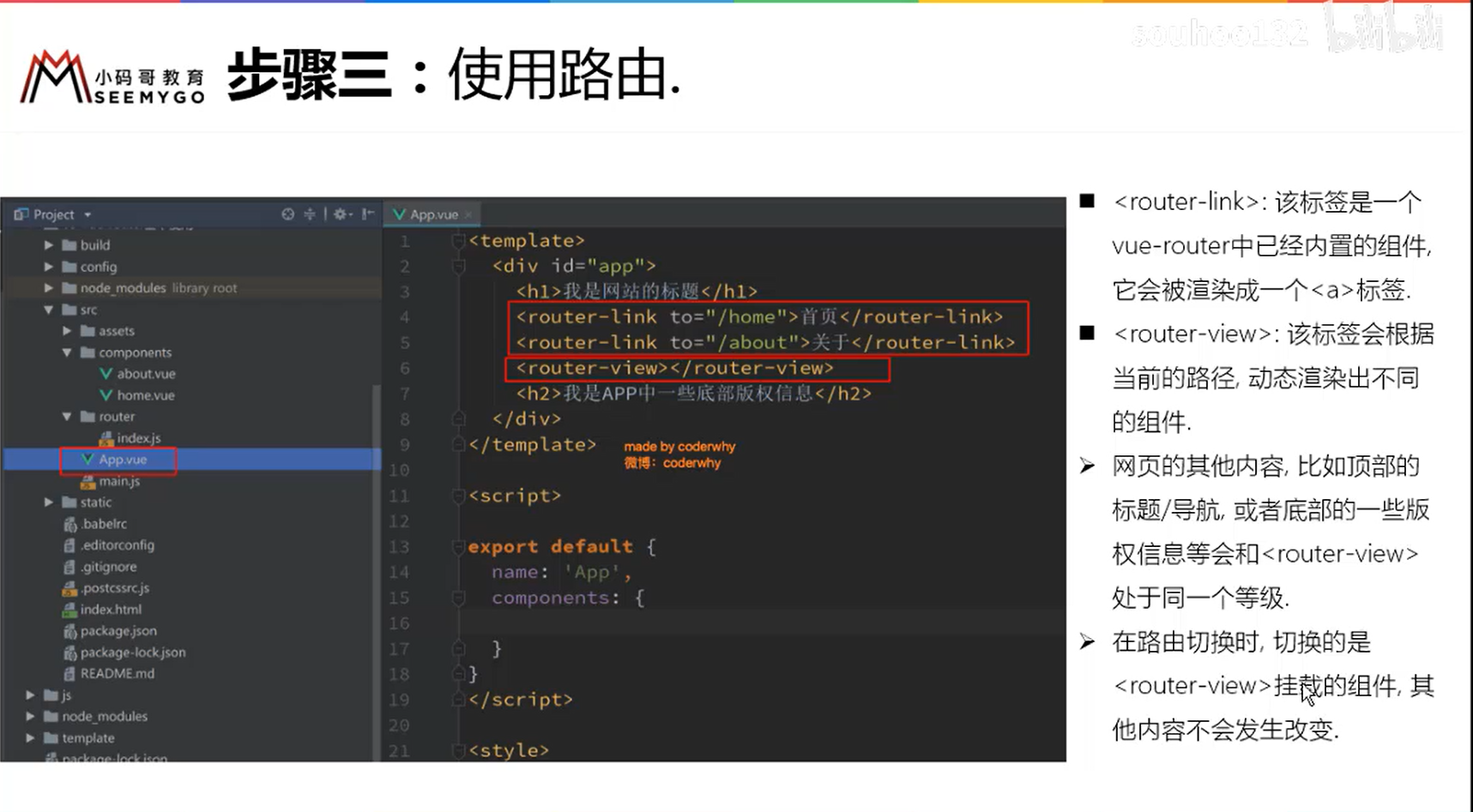
**使用过程:**

配置好路由以后，要在展示入口文件中使用，展示的入口文件即App.vue

因为App.vue是main.js项目入口文件 唯一挂载的vue组件。

一、<router-link to=””> </router-link>

二、<router-view></router-view>



一、<router-link to=””> </router-link>

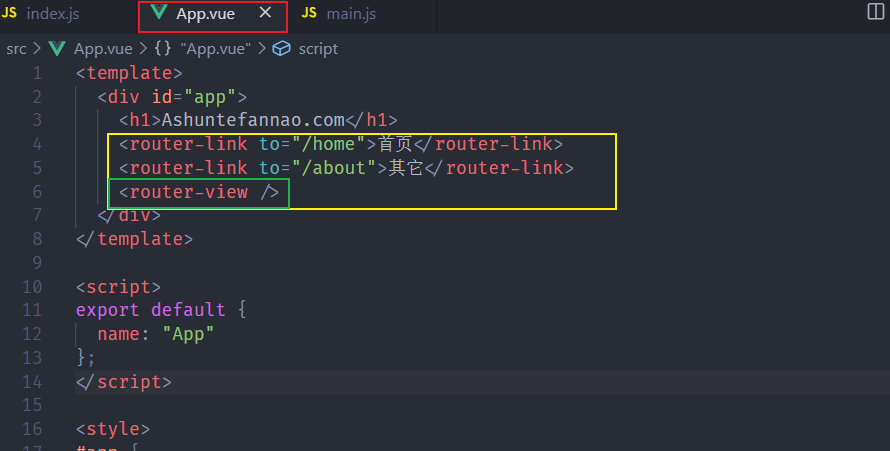
该标签相当于 html中的a标签，最终也会被渲染为a标签。

to=”/path”属性是配置请求的path

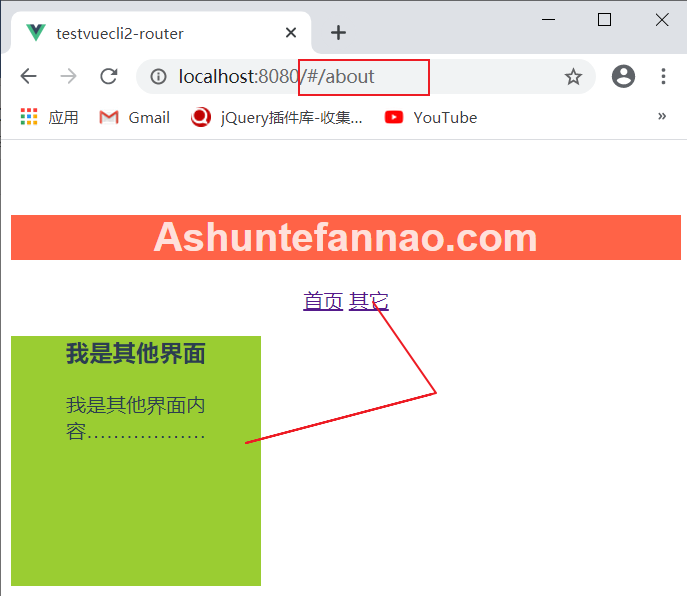
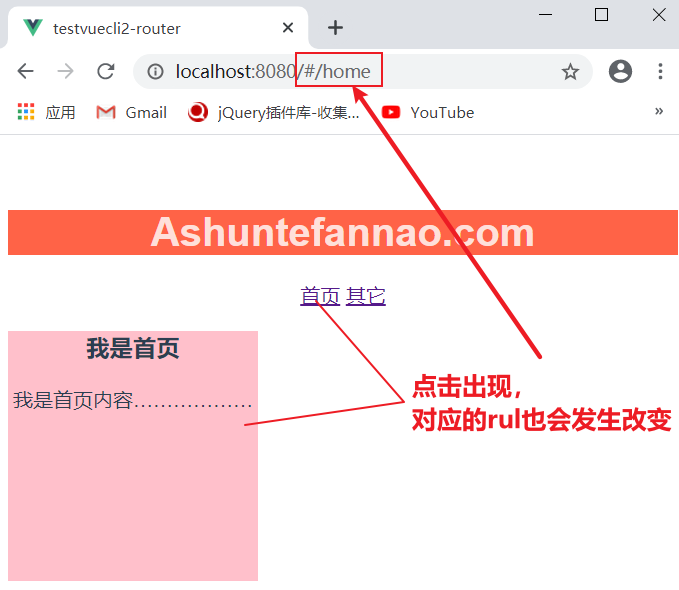
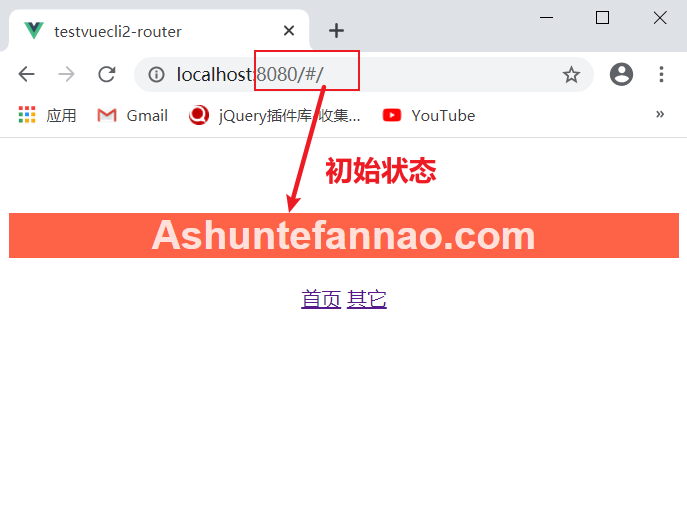
二、<router-view></router-view>

用来展示所抽离的组件，是一个占位标签

该标签在html中的位置，决定了展示时候的位置

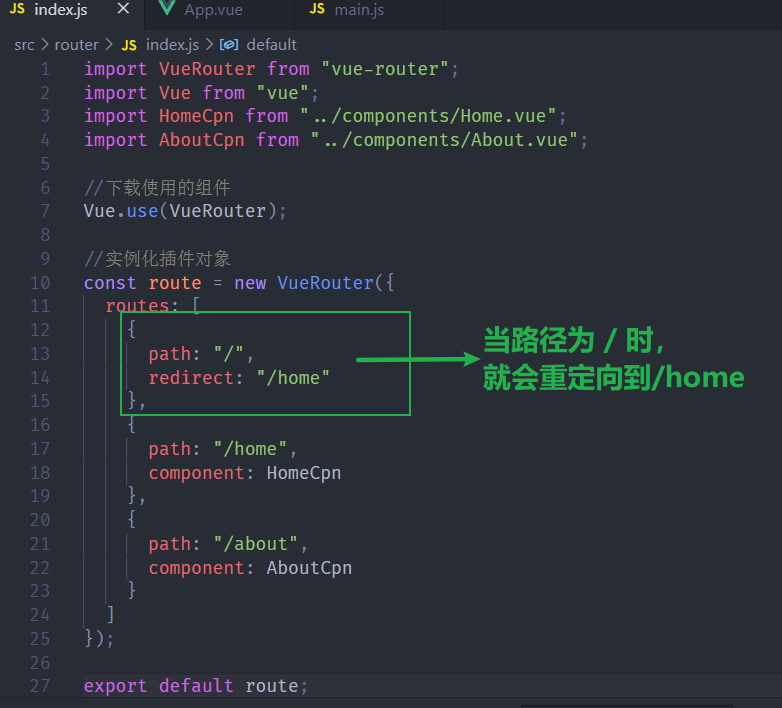


**效果：**

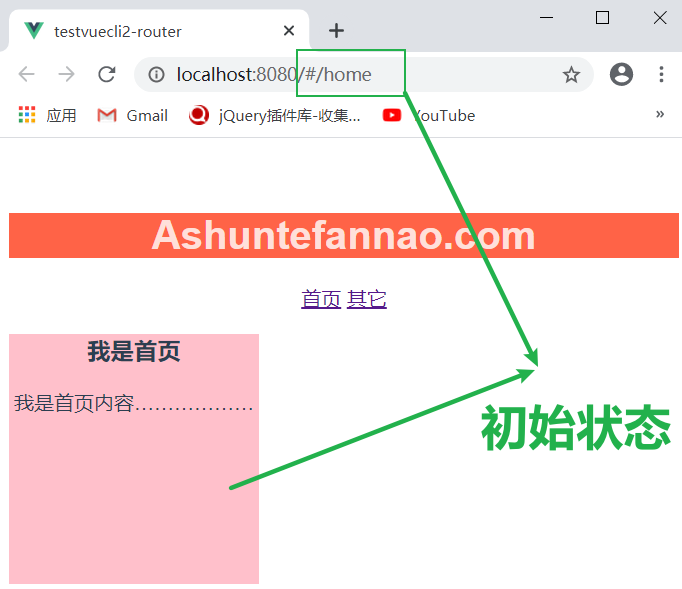


**设置默认路由“/”，并配置 重定向redirect：**

若想要一开始就展示首页面，只要配置默认地址的路由”/”即可，但是点击首页后，连接会更改为home，导致连接的不一致，这时候可以使用redirect重定向



**效果：**

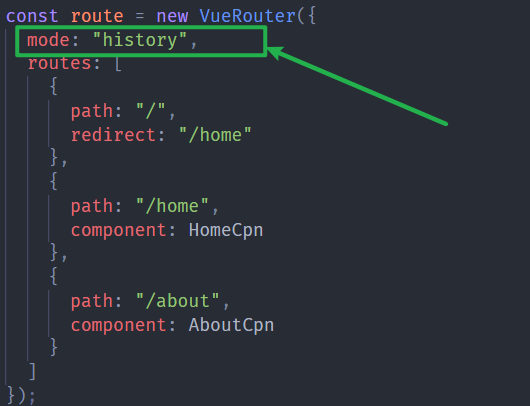


**更改url为history模式**

我们发现vue-router默认使用location.hash管理前端路由，但是更改hash，请求的path前面会多一个 “#” 很不美观，也不像一个真正的url。

**可以在vue-router实例化类的参数对象中配置mode (模式)进行更改**

**mode：“history“**



**效果:**

**请求的path前没有了 “#”**

