

# 路由的基本使用

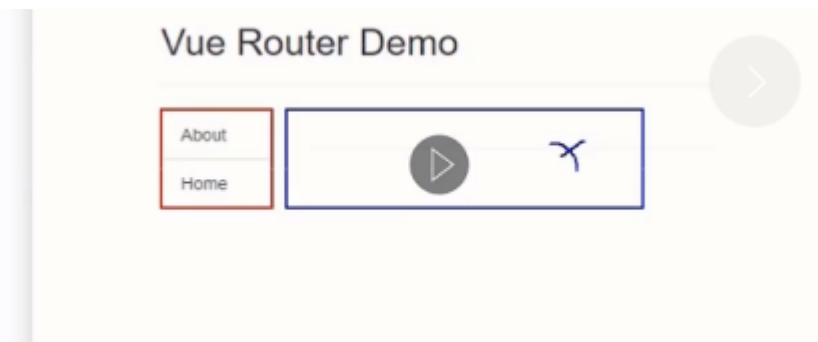
## 基本概念

1. vue-router: vue 的一个插件库，专门用来实现 SPA 应用
2. 对 SPA 应用的理解
  1. 单页 Web 应用 ( single page web application, SPA) 。
  2. 整个应用只有一个完整的页面。
  3. 点击页面中的导航链接不会刷新页面，只会做页面的局部更新。
  4. 数据需要通过 ajax 请求获取。
3. 什么是路由：
  1. 一个路由就是一组映射关系 ( key - value)
  2. key 为路径, value 可能是 function 或 component
4. 前端路由：
  1. 理解：value 是 component，用于展示页面内容。
  2. 工作过程：当浏览器的路径改变时, 对应的组件就会显示。

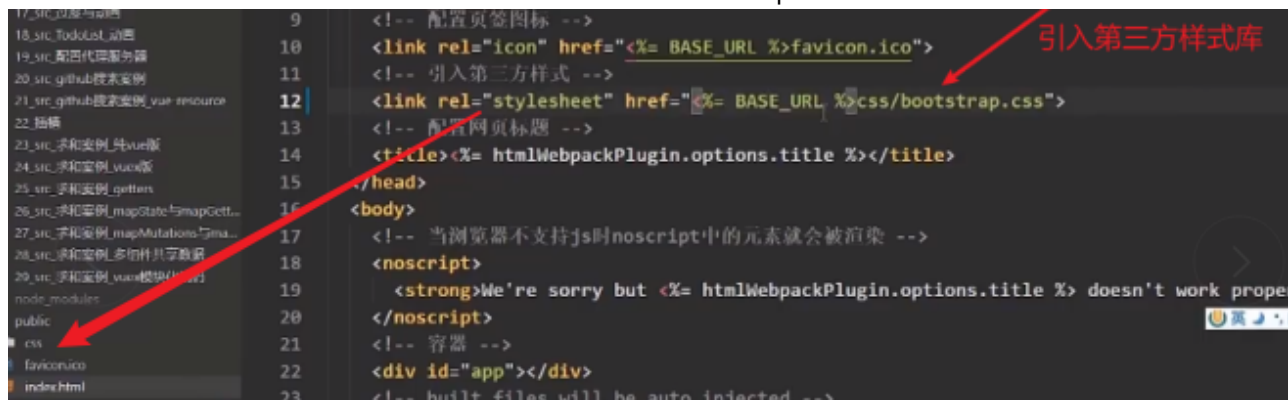
## 编写路由步骤

1. 配置文件index
2. 定义路由组件
3. 注册路由组件
4. 使用路由

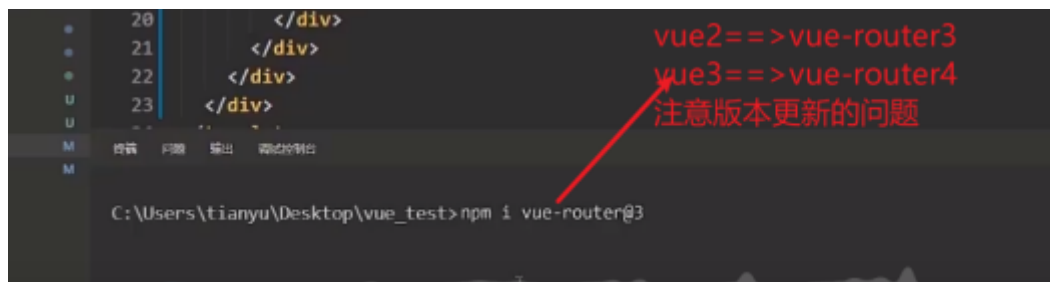
## 案例引出



- 前期准备，静态页面提前准备好，这里的CSS样式引入bootstrap



- router插件的引入，此处注意版本



## 配置文件index

```
// 该文件用于配置整个应用的路由器
import VueRouter from 'vue-router'

// 引入路由组件
import About from '../components/About'
import Home from '../components/Home'

export default new VueRouter({
  routes:[
    {
      path: '/about',
      component: About
    },
    {
      path: '/home',
      component: Home
    }
  ]
})
```

- 在主文件里引入router并声明对象

```
import Vue from "vue"
import App from "./App.vue"

// 引入路由器
import VueRouter from 'vue-router'
import router from './router/'
Vue.use(VueRouter)

new Vue({
  render: h => h(App),
  router:router
}).$mount('#app')
```

## 定义路由组件

- 这里定义路由组件跟普通组件的定义一样，开发的时候会新建一个文件夹pages 默认里面存放的是路由组件

## 路由组件的注册

- 路由数组注册在VueRouter对象里面的routers 的对象数组里面

```
export default new VueRouter({
  routes:[
    {
      path:'/about',
      component: About
    },
    {
      path:'/home',
      component:Home
    }
  ]
})
```

## 路由的使用

- 路由组件在父组件里面的声明---就是规定路由跳转的路径
  - 路由组件在父组件里面的展示---就是当跳转到指定路径时，在父组件里面的路由组件展示的地方
- router-link to ---本质还是生成了一个a标签的链接

```
<!-- 原始html使用 a 链接实现页面的跳转 -->
<!-- <a class="list-group-item" href="./about.html">About</a>
<a class="list-group-item active" href="./home.html">Home</a> -->

<!-- Vue 中借助router-link标签实现路由的切换 -->
<router-link class="list-group-item" active-class="active"
to="/about">About</router-link>
<router-link class="list-group-item" active-class="active"
to="/home">Home</router-link>
```

### ◦ router-view

```
<div class="panel-body">
  <router-view></router-view>
</div>
```

## 几个注意点

## 2.几个注意点

1. 路由组件通常存放在 `pages` 文件夹，一般组件通常存放在 `components` 文件夹。
2. 通过切换，“隐藏”了的路由组件，默认是被销毁掉的，需要的时候再去挂载。
3. 每个组件都有自己的 `$route` 属性，里面存储着自己的路由信息。
4. 整个应用只有一个 `router`，可以通过组件的 `$router` 属性获取到。

## 嵌套（多级）路由

### 二级路由的配置文件index

```
export default new VueRouter({
  routes: [
    {
      path: '/about',
      component: About
    },
    {
      path: '/home',
      component: Home,
      children: [
        {
          path: 'message',
          component: Message
        },
        {
          path: 'news',
          component: News
        }
      ]
    }
  ]
})
```

### 二级路由组件的定义

- 二级路由或者是多级路由组件的定义都在 `pages` 路由组件文件里面定义，跟一般组件的定义一致

### 二级路由组件的注册

- 二级路由组件在配置文件里面的注册写在上一级路由组件的配置里面

```
children: [  
  {  
    path: 'message',  
    component: Message  
  },  
  {  
    path: 'news',  
    component: News  
  }  
]
```

- 注意多级路由的路径里面不需要加 / 系统自动声明

## 多级路由组件的使用

- 多级路由的路径配置和展示配置都要写在上一级路径的组件里面
  1. 配置路径

```
<ul class="nav nav-tabs">  
  <li>  
    <!-- <a class="list-group-item" href="./home-news.html">News</a> -->  
    <router-link class="list-group-item" active-class="active"  
to="/home/news">News</router-link>  
  </li>  
  <li>  
    <!-- <a class="list-group-item active" href="./home-  
message.html">Message</a> -->  
    <router-link class="list-group-item" active-class="active"  
to="/home/message">Message</router-link>  
  </li>  
</ul>
```

### 2. 配置展示

```
<div>  
  <router-view></router-view>  
</div>
```