

link: null
title: 珠峰架构师成长计划
description: 引入bootstrap,进行安装
keywords: null
author: null
date: null
publisher: 珠峰架构师成长计划
stats: paragraph=43 sentences=87, words=721

vue路由使用详解#

基础的路由配置#

```
import Vue from 'vue'
import Router from 'vue-router'
import routes from './routes';
Vue.use(Router)

export default new Router({
  mode: 'history', // &#x4E0D;&#x4F7F;&#x7528;#&#x65B9;&#x5F0F;
  base: process.env.BASE_URL,
  routes
});

// routes.js&#x6587;&#x4EF6;&#x662F;&#x4E13;&#x95E8;&#x5904;&#x7406;&#x8DEF;&#x7531;&#x7684; &#x914D;&#x7F6E;&#x5F71;&#x5C04;&#x5173;&#x7CFB;
export default [
  {
    path: '/',
    redirect: {name: 'home'} // &#x9ED8;&#x8BA4;&#x8BBF;&#x95EE;&#x8839;&#x8DEF;&#x5F84;&#x65F6;
&#x53EF;&#x4EE5;&#x91CD;&#x5B9A;&#x5411;&#x5230;home&#x8DEF;&#x7531;
  },
  {
    name: 'home',
    path: '/home', // &#x5B9E;&#x73B0;&#x4E86;&#x8DEF;&#x7531;&#x7684;&#x61D2;&#x52A0;&#x8F7D;
    component: () => import ('../views/Home.vue')
  },
  {
    path: '/profile',
    name: 'profile',
    component: () => import ('../views/Profile.vue')
  },
  {
    path: '/user',
    name: 'user',
    component: () => import ('../views/User.vue')
  },
  {
    path: '/login',
    name: 'login',
    component: () => import ('../views/Login.vue')
  },
  {
    path: '*',
    component: () => import ('../views/404.vue')
  }
]
```

路由的导航应用#

引入bootstrap,进行安装

```
yarn add bootstrap@3
import 'bootstrap/dist/css/bootstrap.css'
```

```
<template>
<div>
  <nav class="navbar navbar-default">
    <div class="container-fluid">
      <div class="navbar-header">
        <a class="navbar-brand" href="#">
          &#x7BA1;&#x7406;&#x7CFB;&#x7EDF;
        </a>
      </div>
      <ul class="navbar-nav nav">
        <li>
          <router-link to="{name: 'home'}">&#x9996;&#x9875;</router-link>
        </li>
        <li>
          <router-link to="{name: 'profile'}">&#x4E2A;&#x4EBA;&#x4E2D;&#x5FC3;</router-link>
        </li>
        <li>
          <router-link to="{path: '/user'}">&#x7528;&#x6237;&#x7BA1;&#x7406;</router-link>
        </li>
        <li>
          <router-link to="{name: 'login'}">&#x767B;&#x5F55;</router-link>
        </li>
      </ul>
    </div>
  </nav>
</div>
</template>
```

配置命名视图 (一个路径匹配多个路由)#

在项目中一般情况是不使用命名视图的

```
{
  name: 'home',
  path: '/home', //  &#x5B9E;&#x73B0;&#x4E86;&#x8DEF;&#x7531;&#x7684;&#x61D2;&#x52A0;&#x8F7D;
  components: { //  &#x5F53;&#x524D;&#x8DEF;&#x5F84;  &#x5339;&#x914D;&#x5BF9;&#x4E2A;router-view
    default: () =>import ('../views/Home.vue'),
    name: () =>import ('../components/Name.vue'),
    version: () =>import ('../components/Version.vue')
  }
}
```

二级路由配置

父路径跳转必须使用路径跳转使用nam, 不显示默认孩子

```
{
  path: '/user',
  name: 'user',
  component: () =>import ('../views/User.vue'),
  children: [ //  &#x5B50;&#x8DEF;&#x7531;&#x914D;&#x7F6E;
    {
      path: '', //  &#x5982;&#x679C;&#x6E32;&#x67D3;&#x9ED8;&#x8BA4;&#x8DEF;&#x7531;  &#x5FC5;&#x987B;&#x4F7F;&#x7528;path&#x8DF3;&#x8F6C;
      redirect: {name: 'userAdd'}
    },
    {
      name: 'userAdd',
      path: 'userAdd',
      component: () =>import ('../views/UserAdd')
    },
    {
      name: 'userList',
      path: 'userList',
      component: () =>import ('../views/UserList')
    }
  ]
},
```

二级路由跳转

```
// user.vue
<template>
<div class="row">
  <div class="col-md-3">
    <nav class="nav nav-stacked">
      <li>
        <router-link :to="{name: 'userAdd'}">&#x7528;&#x6237;&#x6DFB;&#x52A0;</router-link>
      </li>
      <li>
        <router-link :to="{name: 'userList'}">&#x7528;&#x6237;&#x5217;&#x8868;</router-link>
      </li>
    </nav>
  </div>
</div>
</template>
```

编程式导航

```
<template>
  <div>
    <button class="btn btn-primary" @click="login">&#x767B;&#x5F55;</button>
  </div>
</template>

<script>
export default {
  methods: {
    login() {
      this.$router.push ({name: 'login'})
    }
  },
}
</script>
```

组件内的钩子

```
beforeRouteLeave (to, from, next) {
  if(this.username){
    let confirm = window.confirm('&#x786E;&#x8BA4;&#x79BB;&#x5F00;&#x5417;');
    if(confirm) return next();
  }else{
    next();
  }
}

beforeRouteEnter(to,from,next){
  //  &#x5728;&#x8FDB;&#x5165;&#x8DEF;&#x7531;&#x4E4B;&#x524D;&#x53EF;&#x4EE5;&#x5BF9;&#x6570;&#x636E;&#x6821;&#x9A8C;  ajax&#x83B7;&#x53D6;
  let user = JSON.parse(localStorage.getItem('user')) || [];
  if(user.length){
    next();
  }else{
    next ({name: 'userAdd'})
  }
},
```

带参数跳转

```
{
  name: 'userDetail',
  path: 'userDetail',
  component: () => import('../views/UserDetail')
},
{
  name: 'detail',
  path: 'userDetail/:name', // 8DEF;5F84;53C2;6570; 5FC5;987B;901A;8FC7;name8DF3;8F6C;
  component: () => import('../views/UserDetail')
}
// 4E0D;540C;65B9;5F0F;7684;4F20;9012;53C2;6570;
<td><router-link to="{name:'userDetail',query:{id:u.id}}">{{u.id}}</router-link></td>
<td><router-link to="{name:'detail',params:{name:u.username}}"> {{u.username}}</router-link></td>

// 53D6;503C;65B9;5F0F;
67E5;8BE2;5B57;7B26;4E32; {{this.$route.query && this.$route.query.id}}
67E5;8BE2;53C2;6570; {{this.$route.params && this.$route.params.name}}
```

权限校验

```
router.beforeEach((to, from, next) => {
  let needLogin = to.matched && to.matched.some(({meta}) => {
    return meta && meta.needLogin
  });
  let isLogin = localStorage.getItem('login')
  if(needLogin){
    if(isLogin){
      next();
    }else{
      next({name: 'login'});
    }
  }else{
    // 5982;679C;4E0D;9700;8981;767B;5F55; 5E76;4E14;662F;767B;5F55;9875;9762;
    if(!(to.name === 'login' && isLogin)){
      next();
    }
  }
});
```

路由更新变化

```
// watch:{
//   $route(){
//     alert('53C2;6570;53D8;5316;')
//   }
// }
beforeRouteUpdate() { // 5982;679C;7EC4;4EF6;88AB;590D;7528; 552F;4E00;7684;65B9;5F0F;
5C31;662F;76D1;542C;8DEF;7531;7684;66F4;65B0;65B9;6CD5;
  console.log('8DEF;7531;66F4;65B0;4E86;')
}
```

钩子执行属性

- 导航被触发。
- 在失活的组件里调用离开守卫。
- 调用全局的 `beforeEach` 守卫。
- 在重用的组件里调用 `beforeRouteUpdate` 守卫 (2.2+).
- 在路由配置里调用 `beforeEnter`。
- 解析异步路由组件。
- 在被激活的组件里调用 `beforeRouteEnter`。
- 调用全局的 `beforeResolve` 守卫 (2.5+).
- 导航被确认。
- 调用全局的 `afterEach` 钩子。
- 触发 DOM 更新。
- 用创建好的实例调用 `beforeRouteEnter` 守卫中传给 `next` 的回调函数。

导航激活样式

```
.router-link-active{
  color:red!important
}
.nav-stacked .router-link-active{
  color:blue !important
}
```

路由动画

```
.fade-enter{ opacity:0}
.fade-enter-active{ transition:all 0.2s linear;}
.fade-enter-to{opacity:1}
.fade-leave{opacity:1}
.fade-leave-active{transition:all .2s linear}
.fade-leave-to{opacity:0}
```

<https://github.com/wakeupmypig/vue-router-apply> (<https://github.com/wakeupmypig/vue-router-apply>)