vue-router中：

........

........

var router = new VueRouter({

mode:'history',

linkActiveClass:'is-active',

routes: [

{

**path: '/home',**

component: home

},

{

**path: '/about',**

component: about

},

{

**path: '/odocument',**

component: odocument

}

]

})

export default router

如果用户访问'/abc'路径，由于上面没有此路径对应的组件。

那么，将不会显示什么。（这样不是很友好）

**可以这样改变**：

当用户方位没有配置的路径时匹配一个404页面

..........

...........

Vue.use(VueRouter)

var router = new VueRouter({

mode:'history',

linkActiveClass:'is-active',

routes: [

{

**path: '/home',**

component: home

},

{

**path: '/about',**

component: about

},

{

**path: '/odocument',**

component: odocument

},

{

**path: '\*',**

component: notFound

}

]

})

export default router

path: '\*' 表示没有匹配的路径，就匹配此路径，并渲染此路径对应的notFound插件。

**上面的方式是在没找到的路径，全部定义到一个404页面**

**接下来就使用重定向-redirect：**

**将不存在的url访问，重新定义到已经存在的页面上。**

..........

...........

var router = new VueRouter({

mode:'history',

linkActiveClass:'is-active',

routes: [

{

**path: '/home',**

component: home,

**name:'home'**

},

{

**path: '/about',**

component: about

},

{

**path: '/odocument',**

component: odocument

},

{

**path: '\*',**

//component: notFound

//重定向

//**redirect:'/home'**

//**redirect:{path:'/home'}**

//**redirect:{name:'home'}**

}

]

})

export default router

重定向后，当访问不存在的url，就会指向home页面。

**最后一种重定向的写法是指向路由的name名字，每个路由都有自己的name名字。**

**2.动态设置重定向目标**

**如果想要根据具体的url来访问，已存在的不同的页面，那么就不能使用上面的写法了。**

..........

...........

var router = new VueRouter({

mode:'history',

linkActiveClass:'is-active',

routes: [

{

**path: '/home',**

component: home,

},

{

**path: '/about',**

component: about

},

{

**path: '/odocument',**

component: odocument

},

{

**path: '\*',**

**redirect:（to）=> {**//to保存着一个对象-访问的路径的路由信息

switch(to.path){

case '/123':

return 'home';

break;

case '/456':

return '/odocument';

break;

default:

return '/about';

}

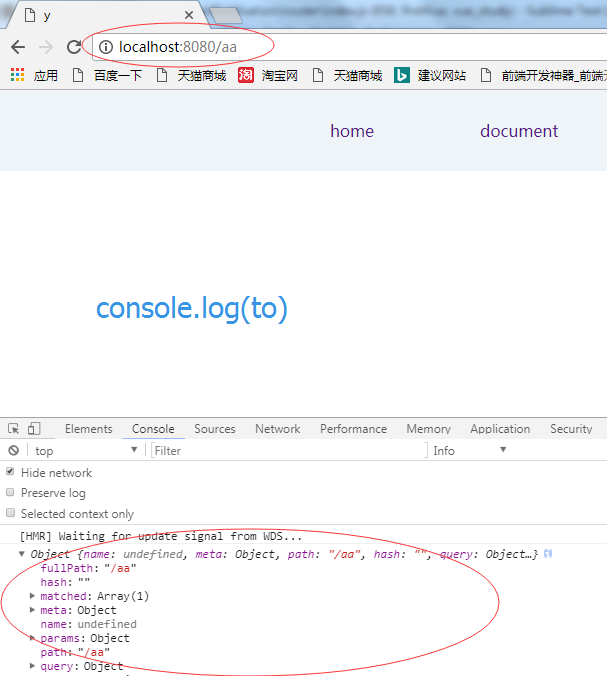
**}**

}

]

})

export default router

****

**3.别名的设置 alias**

..........

...........

var router = new VueRouter({

mode:'history',

linkActiveClass:'is-active',

routes: [

{

**path: '/home',**

component: home,

**alias:'/index'**

},

{

**path: '/about',**

component: about

},

{

**path: '/odocument',**

component: odocument

},

{

**path: '\*',**

component:'notFound'

}

]

})

export default router

**上面代码home设置了别名后，访问 '/index' 页面就会解析 '/home' 对应的组件。**

**注意：虽然访问 '/index' 页面就会解析 '/home' 对应的组件，但是，**

**<router-link>home</router-link>的状态不会被激活。**