使用Query实现路由传参

* **定义路由 router>index.js文件如下：**

1. export default new Router({
2. mode: 'history',
3. routes: [
4. {
5. path: '/one',
6. name: 'ChildOne',
7. component: ChildOne,
8. children:[
9. {
10. path:'/one/log/',
11. component:Log,
12. },
13. {
14. path:'/one/reg/',
15. component:Reg,
16. },
17. ],
18. },
19. ]
20. })

* **修改路由导航 <router-link :to="{path:'/one/log',query:{num:123}}"> ，childOne.vue 文件修改如下：**

1. <template>
2. <div style="border:1px solid red;color:red;">
3. <p>这是父路由childOne对应的组件页面</p>
4. <p>下面可以点击显示嵌套的子路由 </p>
5. <router-link :to="{path:'/one/log',query:{num:123}}">显示登录页面</router-link>
6. <router-link :to="{path:'/one/reg',query:{num:999}}">显示注册页面</router-link>
7. <router-view></router-view>
8. </div>
9. </template>

* **子路由组件通过 this.$route.query.num 来显示传递过来的参数，reg.vue 文件如下：**

1. <template>
2. <div style="border:1px solid purple;color:purple;">
3. <p>注册界面：这是二级路由页面</p>
4. <h3>{{this.$route.query.num}}</h3>
5. </div>
6. </template>

效果如下：  


**PS:**在第一步的定义路由中我们都使用了**mode：history 作用就是去除路由跳转时路由路径前的 “#”**

常用的**mode**模式有两种：

默认为hash模式，最明显的标志是，URL上有#号 localhost:8080/#/，路由会监听#后面的信息变化进行路由匹配

另一种为history模式，不会有#出现，很大程度上对URL进行了美化。需要\*\*注意\*\*history模式在打包后的路由跳转需要服务器配合

默认不使用mode：history 如下

https://img-blog.csdn.net/20180729000400751?watermark/2/text/aHR0cHM6Ly9ibG9nLmNzZG4ubmV0L3pob3V6dW9sdW8=/font/5a6L5L2T/fontsize/400/fill/I0JBQkFCMA==/dissolve/70