**动态路径：**

**我们经常需要把某种模式匹配到的所有路由，全都映射到同个组件。**

例如，我们有一个 User 组件，对于所有 ID 各不相同的用户，都要使用这个组件来渲染。

那么，我们可以在 vue-router 的路由路径中使用『动态路径参数』（dynamic segment）来达到这个效果：

路径：/user/:userId userId 为动态路径参数// 动态路径参数 以冒号开头

获取参数：**路由信息对象**的params

**对组件的注入：**

**通过在Vue根实例的router配置传入router实例**

**在main.js中**

**.............**

**let vm = new Vue({**

**el: '#app',**

**router,**

**template: '<App />',**

**components: {**

**App**

**}**

**})**

**.........**

**1. $router** （每一个组件中都）可以用$router拿到router实例

**2. $route** 表示当前激活的**路由信息对象**，每个组件实例都有

**3. beforeRouteEnter()** 进入组件前钩子函数

**4. beforeRouteLeave()** 离开组件前钩子函数

**路由信息对象中的参数：**

一个路由信息对象，表示当前激活的路由的状态信息，每次成功的导航后，都会产生一个新的对象。（主要有以下参数）

**path** 字符创，对应当前路由的路径。

**params** 对象，包含动态路由参数。

**query** 对象，URL查询参数。

**hash** 字符创，当前路由的hash值。

**fullPath** 字符创，URL包含查询参数和hash的完整路径。

**matched** 数组，包含当前路由的所有嵌套路径片段的 路由记录。

**name** 字符创，当前路由的名称。

**路由中：**

..................

{

path: '/user/**:userId?**',//动态路径的写法 **问号**表示匹配一次或者零次

当匹配到一个路由时，参数值会被设置到 this.$route.params

component:user

}

...................

**user.vue中**

...............................

<template>

<div class='user-list'>

这是user组件

<div>

<router-link **:to**="'/user/'+item.id" key='index' **v-for**="item,index in userList">{{item.userName}}</router-link>

</div>

</div>

</template>

<script>

let data=[

{

id: 1,

userName: '张三',

sex: '男',

hobby: '唱歌'

},

{

id: 2,

userName: '李四',

sex: '男',

hobby: '吃鸡'

},

{

id:3,

userName: '王麻子',

sex: '男',

hobby: '打麻将'

}

]

export default {

data (){

return {

userList:data

}

},

**created** (){//钩子函数

console.log(this.**$route**)

}

}

</script>

........................

