

09. 重定向和组件传参

学习要点：

1. 重定向
2. 组件传参

本节课我们来开始了解 **Vue router** 的重定向和组件传参。

一. 重定向

1. 使用 **redirect**，可以配置路由的重定向功能，有多种方式：

```
{
  //当地址为/a 则跳转到首页
  path : '/a',
  redirect : '/'
},

{
  //跳转到指定 name 的组件
  path : '/a',
  redirect : {
    name : 'List' //或者 path 也支持
  }
},
```

2. 如果你要动态的处理跳转的路径，也就是支持逻辑判断，具体如下：

```
//跳转到闭包
path : '/a',
redirect : () => {
  if (true) {
    return '/list'
  }
}

{
  //跳转到闭包
  path : '/a',
  redirect : to => {
    //获取相关数据
    console.log(to)
    return {
      name : 'List'
    }
  }
}
```

3. 路由可以设置别名，具体如下：

```
{
  path: '/list',
  name: 'List',
  alias : '/liebiao',
}
```

二. 组件传参

1. 需求：我们希望 User/5 中获取 id 时变成 {{id}} 而不是 {{ \$route.params.id }}；
2. 有几种模式，先采用布尔模式，将路由规则增加一个属性：props:true；

```
//router/index.js
{
  path: '/user/:id',
  name: 'User',
  component: User,
  //这里布尔为 true 时，route.params 变成组件属性
  props : true,
}
```

```
//User.vue
<template>
  <div>
    <h3>User Id : {{id}}</h3>
    <router-view></router-view>
  </div>
</template>

<script>
  export default {
    name: "User",
    props : ['id']
  }
</script>
```

3. 除了布尔模式，还有对于静态路由，可以使用对象模式；

```
//支持非命名视图的对象模式
props : {
  name : '列表'
},

<template>
  <h3>这里是 List 页面 : {{name}}</h3>
</template>
```

```
<script>
  export default {
    name: "List",
    props : {
      name : {
        type : String,
        default : 'no data'
      }
    }
  }
</script>
```

//如果是命名视图，则需要特别支出哪个视图使用

```
props : {
  default : {
    name : '列表'
  }
},
```

4. 如果采用经典的 query 进行传参，我可以使用如下方法：

//非命名视图方案

```
props : route => ({
  search: route.query.search
}),
```

```
<template>
  <div class="about">
    <h1>This is an about page : {{search}}</h1>
  </div>
</template>
```

```
<script>
  export default {
    name: "About",
    props : {
      search : {
        type : String,
        default : 'no data'
      }
    }
  }
</script>
```

```
//非命名视图方案
props : {
  default : route => ({
    search: route.query.search,
  }),
},
```