



链滴

Vue 路由传参三种基本方式

作者: [Lee981265](#)

原文链接: <https://ld246.com/article/1548239501950>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

现有如下场景，点击父组件的li元素跳转到子组件中，并携带参数，便于子组件获取数据。

父组件中：

```
<li v-for="article in articles" @click="getDescribe(article.id)">
```

methods:

case1

```
getDescribe(id) {  
  //直接调用$thisrouter.push 实现携带参数的跳转  
  this.$router.push({  
    path: `/describe/${id}`,  
  })  
}
```

case1需要对应路由配置如下：

```
{  
  path: '/describe/:id',  
  name: 'Describe',  
  component: Describe  
}
```

很显然，需要在path中添加/:id来对应 \$router.push 中path携带的参数。在子组件中可以使用来获取传递的参数值。

```
$route.params.id
```

case2

父组件中：通过路由属性中的name来确定匹配的路由，通过params来传递参数。

```
this.$router.push({  
  name: 'Describe',  
  params: {  
    id: id  
  }  
})
```

对应路由配置: 注意这里不能使用/:id来传递参数了，因为父组件中，已经使用params来携带参数了。

```
{  
  path: '/describe',  
  name: 'Describe',  
  component: Describe  
}
```

子组件中: 这样来获取参数

```
$route.params.id
```

case3

父组件：使用path来匹配路由，然后通过query来传递参数
这种情况下 query传递的参数会显示在url后面?id=?

```
this.$router.push({  
  path: '/describe',  
  query: {  
    id: id  
  }  
})
```

对应路由配置：

```
{  
  path: '/describe',  
  name: 'Describe',  
  component: Describe  
}
```

对应子组件：这样来获取参数

```
$route.query.id
```