title	type	order
Routing	guide	20

Official Router

官方路由

For most Single Page Applications, it's recommended to use the officially-supported vue-router library. For more details, see vue-router's documentation.

一般情况下，建议使用官方提供的路由vue-router。有关详细信息，请参阅文档。

Simple Routing From Scratch

## 开始建立简单的路由
If you just need very simple routing, you can do so by dynamically rendering a page-level component like this:

如果你只需要一个简单的路由，你可以这样写：

```
const NotFound = { template: '<p>Page not found</p>' }
const Home = { template: '<p>home page</p>' }
const About = { template: '<p>about page</p>' }

const routes = {
  '/': Home,
  '/about': About
}
```

```
new Vue({
  el: '#app',
  data: {
    currentRoute: window.location.pathname
  },
  computed: {
    ViewComponent () {
      return routes[this.currentRoute] || NotFound
    }
  },
  render (h) { return h(this.ViewComponent) }
})
```

Combined with the HTML5 History API, you can build a very basic but fully-functional client-side router. To see that in practice, check out this example app.

结合HTML5 History API，你可以构建一个小而全的客户端路由。查看实例，点这里.

Integrating 3rd-Party Routers

整合第三方路由

If there's a 3rd-party router you prefer to use, such as Page.js or Director, integration is similarly easy. Here's a complete example using Page.js.

若要改用第三方路由，比如Page.js 或者 Director, 可以参考这段示例。 下面是使用page.js实现路由的一个完整实例。