

嵌套路由:是路由的多层嵌套。

第一步: 需要在 一个被渲染的组件中嵌套 `<router-view>` 组件。

例如, 在 `User` 组件的模板添加一个 `<router-view>`:

```
const User = {
  template: `
    <div class="user">
      <h2>User</h2>
      <router-view></router-view>
    </div>
  `
}
```

第二步: 在嵌套的出口中渲染组件, 在 `VueRouter` 的参数中使用 `children` 配置:

```
const router = new VueRouter({
  routes: [
    {
      path: '/user/:id',
      component: User,
      children: [
        {
          // 当 /user/:id/profile 匹配成功,
          // UserProfile 会被渲染在 User 的 <router-view> 中
          path: 'profile',
          component: UserProfile
        },
        {
          // 当 /user/:id/posts 匹配成功
          // UserPosts 会被渲染在 User 的 <router-view> 中
          path: 'posts',
          component: UserPosts
        }
      ]
    }
  ]
})
```

注意: `children` 配置就是像 `routes` 配置一样的路由配置数组, 所以呢, 你可以嵌套多层路由。

注意：如果基于上面的配置，当你访问 `/user/foo` 时，`User` 的出口是不会渲染任何东西，这是因为没有匹配到合适的子路由。如果你想要渲染点什么，可以提供一个 空的 子路由：

```
const router = new VueRouter({
  routes: [
    {
      path: '/user/:id',
      component: User,
      children: [
        // 当 /user/:id 匹配成功，
        // UserHome 会被渲染在 User 的 <router-view> 中
        { path: '', component: UserHome },

        // ... 其他子路由
      ]
    }
  ]
});
```