

## 13. 导航守卫

学习要点：

### 1. 导航守卫

本节课我们来开始了解 Vue router 的导航守卫的功能。

#### 一. 导航守卫

1. 导航守卫的作用就是跳转或取消跳转的导航功能，比如登录跳转方案；
2. 首先，第一个先来看下全局前置守卫跳转，这里采用新的项目创建；
3. 创建一个 `Login.vue` 的视图组件，并配置路由，用于未登录状态的跳转；
4. 在路由对象声明后，执行全局前置守卫，就是在跳转之前判断操作方案；

```
//全局前置守卫
router.beforeEach((to, from, next) => {
  ...
})
```

PS：关于三个参数的作用如下：

- (1) `.to`：进入目标的路由对象；
- (2) `.from`：正要离开的路由对象；
- (3) `.next`：这是个钩子函数，具体用法下面会说；

5. 模拟一个登录判断，先来一个错误例子，无限执行递归的问题；

```
//模拟登录状态
const flag = true

//全局前置守卫
router.beforeEach((to, from, next) => {
  //如果已登录则不要拦截
  if (flag) {
    //继续执行跳转操作
    next()
  } else {
    //跳转到 login 页面，但会无限递归错误
    next('/login') //或 next({name : 'Login'})
  }
})
```

6. 解决无限递归问题，就是判断目标对象是否是 `login` 即可；

```
//判断是否跳转到 login
if (to.name === 'Login') next()
//跳转到 login
else next('/login')
```

7. 登录状态，不可以进登录页面，让其进首页；

```
//登录状态再跳转 login , 让其直接进首页
if (to.name === 'Login') next('/')
//继续执行跳转操作
else next()
```

8. 全局后置钩子，一般用于路由页面加载完毕之后操作一些动作，比如取消 loading；

```
//全局后置钩子
router.afterEach((to, from) => {
  console.log('关闭 loading...')
})
```