**1.组件中的导航钩子函数beforeRouteEnter：**

例如：在about组件中

......................

<template>

<div>

关于我！

<hr>

<ul class='nav about-nav'>

<router-link to="/about" exact tag='li'>

<a>study</a>

</router-link>

<router-link :to="{name:'Work'}" tag='li'>

<a>work</a>

</router-link>

<router-link to="/about/hobby" tag='li' >

<a>hobby</a>

</router-link>

</ul>

<hr>

<router-view calss='tow-class'></router-view>

</div>

</template>

<script>

export default {

**beforeRouteEnter**(to,from,next){

console.log('beforeRouteEnter')

**next()**

}

}

</script>

.......................................

上面代码表示进入about组件前执行的操作,在切换到about组件时，只会打印'beforeRouteEnter'，只有在执行**next()**之后才会进入该组件。

**beforeRouteEnter中要注意的this问题**

有这样一种场景，组件中有一些数据，执行beforeRouteEnter时（进入组件前）来改写这些数据

例：在about组件中

............................

<template>

<div>

关于我！

<hr>

<ul class='nav about-nav'>

<router-link to="/about" exact tag='li'>

<a>study</a>

</router-link>

<router-link :to="{name:'Work'}" tag='li'>

<a>work</a>

</router-link>

<router-link to="/about/hobby" tag='li' >

<a>hobby</a>

</router-link>

</ul>

数据测试：{{**test**}}

<hr>

<router-view calss='tow-class'></router-view>

</div>

</template>

<script>

export default {

data(){

return {

**test**:'修改前'

}

},

beforeCreate(){//组件创建时最先执行的钩子函数

console.log('beforeCreate')

},

beforeRouteEnter(to,from,next){

console.log('beforeRouteEnter')

next()

console.log(this)//undefined

}

}

</script>

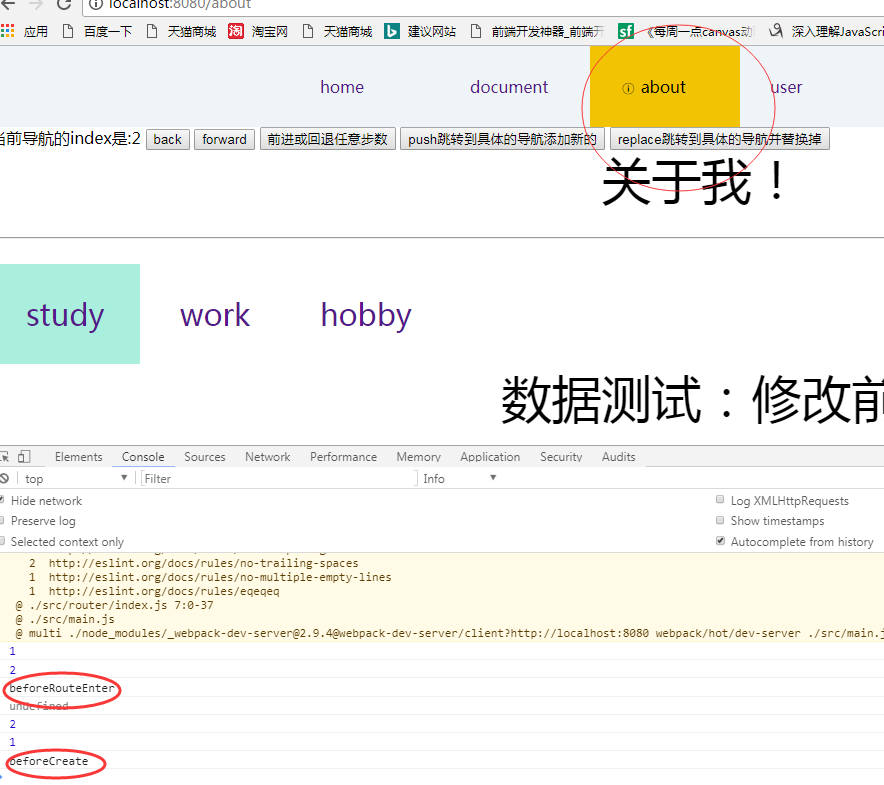
....................

上面代码在beforeRouteEnter中打印this显示undefined，原因是在执行beforeRouteEnter这个方法时，这个about组件实例还没有被创建。

可以从beforeCreate方法和beforeRouteEnter方法中打印出字符串的先后顺序可以看出来，

'beforeRouteEnter'字符串先打印，'beforeCreate'字符串后打印，所以beforeRouteEnter方法先执行，而beforeCreate方法是组件创建后最先执行的钩子函数。

这就是上面beforeRouteEnter方法中为什么**aboout的组件实例this**显示undefined的原因了。



**怎样解决上面的问题呢？**

beforeRouteEnter方法的的next()方法可以接收一个回调函数，

可以将vm（组件实例）传到next的回调函数里，这样就解决了。

例：about组件中：

......................

beforeRouteEnter(to,from,next){

console.log('beforeRouteEnter')

**next((vm)=>{**

**console.log(vm.test)**//'修改前'

**vm.test="改变了"**//改变了about组件里test的值

**})**

console.log(this)//undefined

}

.................

**上面next()回调中的vm参数就表示当前组件的实例**

**2.组件中的导航钩子函数beforeRouteUpdate：**

在有二级导航的组件里，如上面的about组件，由于about组件只被初始化一次，

二级导航组件只是复用about组件,那么在切换二级导航的时候，怎么才能知道呢？

二级组件的变化（点击、切换）即更新，就会触发导航钩子函数beforeRouteUpdate

例：在about组件中

.................

beforeRouteUpdate(to,from,next){

console.log('beforeRouteUpdate')

next()

}

...................

上面的代码，在切换about的二级导航时，都会打印出'beforeRouteUpdate'字符串，但是，要进入具体二级组件（显示对应的内容），就需要执行next()方法。

**3.组件中的导航钩子函数beforeRouteLeave：**

如果不想离开当前组件如上面的about组件，那么可以嗲用beforeRouteLeave方法

不执行next()方法

例：在about组件中

.................

beforeRouteLeave(to,from,next){

console.log(beforeRouteLeave)

//next()//当不执行此方法时表示不离开当前about组件,即切换不了

}

...................