**再定义一个左切换的动画**

在App.vue中：

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

<router-view name="doc\_slider"></router-view>

<transition name="left">

<router-view class='center'></router-view>

</transition>

<router-view name='doc\_footer'></router-view>

</div>

</template>

<script>

export default {

name: 'app',

data (){

return {

index: '/'

}

}

}

</script>

<style>

.v-enter{

opacity:0;

}

.v-enter-to{

opacity:1;

}

.v-enter-active{

transition:1s;

}

.v-leave{

opacity:1;

}

.v-leave-to{

opacity:0;

}

.v-leave-active{

transition:1s;

}

**.left-enter**{

transform: translateX(100%);

}

**. left -enter-to**{

transform: translateX(0);

}

**. left -enter-active**{

transition:2s;

}

**. left -leave**{

transform: translateX(-100%);

}

**. left -leave-to**{

transform: translateX(0);

}

**. left -leave-active**{

transition:1s;

}

</style>

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

**将<transition>的name,设置成left**

**再定义一个向右切换的动画**

在App.vue中：

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

<router-view name="doc\_slider"></router-view>

<transition name="left">

<router-view class='center'></router-view>

</transition>

<router-view name='doc\_footer'></router-view>

</div>

</template>

<script>

export default {

name: 'app',

data (){

return {

index: '/'

}

}

}

</script>

<style>

.v-enter{

opacity:0;

}

.v-enter-to{

opacity:1;

}

.v-enter-active{

transition:1s;

}

.v-leave{

opacity:1;

}

.v-leave-to{

opacity:0;

}

.v-leave-active{

transition:1s;

}

//向左的动画

**.left-enter**{

transform: translateX(100%);

}

**. left -enter-to**{

transform: translateX(0);

}

**. left -enter-active**{

transition:2s;

}

**. left -leave**{

transform: translateX(-100%);

}

**. left -leave-to**{

transform: translateX(0);

}

**. left -leave-active**{

transition:1s;

}

//向右的动画

**.right-enter**{

transform: translateX(-100%);

}

**. right -enter-to**{

transform: translateX(0);

}

**. right -enter-active**{

transition:2s;

}

**. right -leave**{

transform: translateX(0);

}

**. right -leave-to**{

transform: translateX(100%);

}

**. right -leave-active**{

transition:1s;

}

</style>

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

左右动画都实现了怎么实现点击左边用向左切换动画，点击右边用向右切换动画？

这里需要动态的改变**<transition>**的name的值

可以用路由的meta（路由元信息），将需要的信息配置在路由信息对象（$route）的meta对象中.

例如给每个导航栏home、document、about、user添加一个index值

当切换的时候对index进行监控，来决定使用向左还是向右的name名。

用**$route.meta访问meta中的数据**

路由中：

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

routes: [

{

path:'/',

component:home,

**meta:{index:0}**

},

{

path: '/home',

component: home,

name: 'home',

alias:'/index'//为home页设置别名

},

{

path: '/about',

component: about,

children:[

{

path:'',//设置为空默认访问此组件

component:study,

**meta:{index:2}**

},

{

path:'work',

component:work,

name:'Work'

},

{

path:'hobby',

component:hobby

},

]

},

{

path: '/odocument',

components:{//一个路由对应多个组件

default:odocument,//默认组件

doc\_slider:slider,

doc\_footer:footer

},

**meta:{index:1}**

},

{

path: '\*',

//component: notFound

//重定向

//redirect:'/home'

//redirect:{path:'/home'}

//redirect:{name:'home'}

//动态重定向

redirect: (to) => {//to保存着一个对象-访问的路径的路由信息

//console.log(to);

switch(to.path){

case '/123':

return 'home';

break;

case '/456':

return '/odocument';

break;

default:

return '/about';

}

}

},

{

path: '/user/:tip?/:userId?',//动态路径的写法 问号表示匹配一次或者零次

component:user,

**meta:{index:3}**

}

]

})

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

App.vue中：

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

<router-view name="doc\_slider"></router-view>

<transition **:name="dic"**>

<router-view class='center'></router-view>

</transition>

当前导航的index是:{{$route.meta.index}}

<router-view name='doc\_footer' class='footer'></router-view>

</div>

</template>

<script>

export default {

**watch:{**

$route(to,from){

console.log(to.meta.index);//目标导航的下标

console.log(from.meta.index);//离开导航的下标

if(to.meta.index>from.meta.index){

this.dic="right"

}else{

this.dic="left"

}

}

},

name: 'app',

data (){

return {

index: '/',

**dic:'left'**

}

}

}

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

上面代码通过meta为每个导航加上了index，动态绑定transition标签的name的值。

用watch监控$route.meta，to，from参数的变化即是index值的变化。

直接通过index值大小的判断来设置动态的name值。