

路飞学城项目前端

路飞学城项目前端

笔记

5. 免费课

5.1 在routers/index.js路由中注册路由

5.2 完成免费课课程列表

5.2.1 App.vue代码,记录初始化的样式

5.2.2 Courses.vue的文件代码结构

5.2.3 courses.vue的css代码,保存课程列表的全部样式

6. 针对页面头部部分增加高亮和登录按钮

6.1 增加当前页面导航高亮效果

6.2 根据登录状态显示登录注册按钮

7. 购物车页面

7.1 创建购物车页面组件

7.2 注册路由地址

7.3 在cart购物车组件中的头部和脚部引入

7.4 把购物车中信息作为子组件,引入到购物车页面中

7.5 在Cart.vue购物车页面组件中引入CartItem商品信息组件

作业讲解

笔记

5. 免费课

在组件目录components下创建Couses.vue组件文件,代码如下:

```
<template>
  <div class="courses">

  </div>
</template>

<script>
  export default {
    name: "Courses",
    data() {
      return {

      }
    }
  }
</script>

<style scoped>
```

```
</style>
```

5.1 在routers/index.js路由中注册路由

```
import Vue from "vue"
import Router from "vue-router"

// 导入可以被用户访问的组件
import Home from "@components/Home"
import Courses from "@components/Courses"

Vue.use(Router);

export default new Router({
  mode: "history",
  routes: [
    // 路由列表
    {
      path: "/",
      name: "Home",
      component: Home,
    },
    {
      path: "/home",
      name: "Home",
      component: Home,
    },
    {
      path: "/courses",
      name: "Courses",
      component: Courses,
    },
  ],
})
```

5.2 完成免费课课程列表

5.2.1 App.vue代码,记录初始化的样式

```
<template>
  <div id="app">
    <router-view/>
  </div>
</template>

<script>
```

```

export default {
  name: 'App',
  components: {
  }
}
</script>

<style>
body,p{
  margin:0;
  padding:0;
}
a{
  color: #4a4a4a;
  text-decoration: none;
}
img{
  width: 100%;
}
</style>

```

5.2.2 Courses.vue的文件代码结构

```

<template>
  <div class="courses">
    <Header/>
    <div class="main">
      <!-- 课程列表代码 -->
    </div>
    <Footer/>
  </div>
</template>

<script>
import Header from "../common/Header"
import Footer from "../common/Footer"
export default {
  name: "courses",
  data() {
    return {
      filter_price: false,
    }
  },
  components: {
    Header,
    Footer,
  },
  methods: {
  }
}

```

```
</script>

<style scoped>
.courses{
  padding-top: 80px;
}
</style>
```

5.2.3 courses.vue的css代码,保存课程列表的全部样式

```
<style scoped>
.courses{
  padding-top: 80px;
}
.main{
  width: 1100px;
  height: auto;
  margin: 0 auto;
  padding-top: 35px;
}
.main .filter{
  width: 100%;
  height: auto;
  margin-bottom: 35px;
  padding: 25px 0px 25px 0px;
  background: #fff;
  border-radius: 4px;
  box-shadow: 0 2px 4px 0 #f0f0f0;
}
.filter .el-col{
  text-align: center;
  padding: 6px 0px;
  line-height: 16px;
  margin-left: 14px;
  position: relative;
  transition: all .3s ease;
  cursor: pointer;
  color: #4a4a4a;
}
.filter-el-row1{
  padding-bottom: 18px;
  margin-bottom: 17px;
}
.filter .filter-text{
  text-align: right;
  font-size: 16px;
  color: #888;
}
.filter .filter-text2{
```

```

}
.filter .filter-el-row1 .current{
    color: #ffc210;
    border: 1px solid #ffc210!important;
    border-radius: 30px;
}
.filter .filter-el-row2 .current{
    color: #ffc210;
}
.filter-price{
    display:inline-block;
    vertical-align: middle;
}
.filter-price .up, .filter-price .down{
    display: block;
    line-height: 8px;
    font-size: 13px;
    margin: -4px;
    color: #d8d8d8;
}
.current .filter-price .active{
    color: #ffc210;
}
.course-item{
    margin-bottom: 35px;
}
.course-item .course-item-box{
    padding: 20px 30px 20px 20px;
}
.course-item{
    box-shadow: 2px 3px 16px rgba(0,0,0,.1);
    transition: all .2s ease;
}
.course-item .course-item-left{
    width: 423px;
    height: 210px;
    margin-right: 30px;
}
.course-title{
    overflow: hidden; /* 在父元素中使用可以清除子元素的浮动影响 */
}
.course-title .box-title{
    font-size: 26px;
    color: #333333;
    float: left;
    margin-bottom: 8px;
}
.course-title .box-number{
    font-size: 14px;
    color: #9b9b9b;
    font-family: PingFangSC-Light;
    float: right;
    padding-top: 12px;
}

```

```
}
.course-item-right{
  width: 56.6%;
}
.author{
  font-size: 14px;
  color: #9b9b9b;
  margin-bottom: 14px;
  padding-bottom: 14px;
  overflow: hidden;
}
.author .box-author{
  float:left;
}
.author .lesson{
  float: right;
}
.course-content .el-icon-caret-right{
  border: 1px solid #000;
  border-radius: 50%;
  margin-right: 6px;
}
.course-content .el-col{
  font-size: 14px;
  color: #666;
  width: 50%;
  margin-bottom: 15px;
  cursor: pointer;
}
.course-content .el-col:hover{
  color: #ffc210;
}
.course-content .el-col:hover .el-icon-caret-right, .course-content .el-col:hover .free{
  border-color: #ffc210;
  color: #ffc210;
}
.course-content .el-col .free{
  width: 34px;
  height: 20px;
  color: #fd7b4d;
  margin-left: 10px;
  border: 1px solid #fd7b4d;
  border-radius: 2px;
  text-align: center;
  font-size: 13px;
  white-space: nowrap;
}
.course-price{
  overflow: hidden;
}
.course-price .course-price-left{
  float: left;
}
```

```

.course-price .discount{
    padding: 6px 10px;
    display: inline-block;
    font-size: 16px;
    color: #fff;
    text-align: center;
    margin-right: 8px;
    background: #fa6240;
    border: 1px solid #fa6240;
    border-radius: 10px 0 10px 0;
}
.course-price .course-price-left{
    line-height: 22px;
}
.course-price .count{
    font-size: 24px;
    color: #fa6240;
}
.course-price .old_count{
    font-size: 14px;
    color: #9b9b9b;
    text-decoration: line-through;
    margin-left: 10px;
}
.course-price .buy{
    float: right;
    width: 120px;
    height: 38px;
    font-size: 16px;
    border-radius: 3px;
    border: 1px solid #fd7b4d;
    background: transparent; /* 透明 */
    color: #fa6240;
    cursor: pointer;
    transition: all .2s ease-in-out; /* css3新版本的样式中支持支持 jQuery里面的动画预设效果 */
    /* all表示当前元素的所有样式 .2s表示改变样式完成的时间 ease-in-out */
}
.course-price .buy:hover{
    color: #fff;
    background: #ffc210;
    border: 1px solid #ffc210;
}
</style>

```

效果:



6. 针对页面头部部分增加高亮和登录按钮

6.1 增加当前页面导航高亮效果

使用current_page的值进行判断是否是当前页面,每次进入页面时,由当前页面的组件设置一个序号值.

Header.vue,视图代码:

```
<el-row class="nav" :gutter="20">
  <el-col :span="3"><router-link :class="current_page==1?'active':''"
to="/courses">免费课</router-link></el-col>
  <el-col :span="3"><router-link :class="current_page==2?'active':''"
to="/courses">轻课</router-link></el-col>
  <el-col :span="3"><router-link :class="current_page==3?'active':''"
to="/courses">学位课</router-link></el-col>
  <el-col :span="3"><router-link :class="current_page==4?'active':''"
to="/courses">题库</router-link></el-col>
  <el-col :span="3"><router-link :class="current_page==5?'active':''"
to="/courses">教育</router-link></el-col>
</el-row>
```

css代码:


```
.nav{
  margin-left: 0!important;
  margin-right: 0!important;
}
.nav .el-col .active{
  padding-bottom: 16px;
  padding-left: 5px;
  padding-right: 5px;
  border-bottom: 4px solid #ffc210;
}
```

效果:



接下来,需要接收到来自于页面组件Home.vue或者Courses.vue的值,所以其实父组件传递数据给子组件.

Header.vue的js代码:

```
<script>
  export default {
    name: "Header",
    props: ["current_page"],
    data(){
      return {
        is_login: false, /* 是否登录 */
      }
    }
  }
</script>
```

效果:

```

55 <script>
56   export default {
57     name: "Header",
58     props: ["current_page"],
59     data() {
60       return {
61         is_login: false, /* 是否登录 */
62       }
63     }
64   }
65 </script>
66

```

子组件接受的current_page是需要在父组件中进行传递的,

Home.vue或者Course.vue以及以后的页面组件.代码:

```

<template>
  <div class="home">
    <Header :current_page="current_page"/>
    <Banner/>
    <Footer/>
  </div>
</template>

<script>
  import Header from "../common/Header"
  import Banner from "../common/Banner"
  import Footer from "../common/Footer"
  export default{
    name: "Home",
    data(){
      return {
        current_page:0,
      };
    },
    components:{
      Header,
      Banner,
      Footer,
    }
  }
</script>

<style scoped>
  .home{
    padding-top: 80px;
  }
</style>

```

效果:

```

2   <div class="home">
3     <Header :current_page="current_page"/>
4     <Banner/>
5     <Footer/>
6   </div>
7 </template>
8
9 <script>
10  import ...
13  export default {
14    name: "Home",
15    data() {
16      return {
17        current_page: 0,
18      };
19    },

```

传递变量使用 :变量="当前父组件中data的数据"

数据声明

效果:



6.2 根据登录状态显示登录注册按钮

```

<template>
  <div class="header">
    <el-container>
      <el-header height="80px">
        <el-row>
          <el-col class="logo" :span="3">
            <router-link to="/"></router-link>
          </el-col>
          <el-col :span="16">
            <!-- gutter每一列之间的间隔空隙 -->
            <el-row class="nav" :gutter="20">
              <el-col :span="3"><router-link :class="current_page==1?'active':''"
to="/courses">免费课</router-link></el-col>
              <el-col :span="3"><router-link :class="current_page==2?'active':''"
to="/courses">轻课</router-link></el-col>
              <el-col :span="3"><router-link :class="current_page==3?'active':''"
to="/courses">学位课</router-link></el-col>
              <el-col :span="3"><router-link :class="current_page==4?'active':''"
to="/courses">题库</router-link></el-col>
              <el-col :span="3"><router-link :class="current_page==5?'active':''"
to="/courses">教育</router-link></el-col>
            </el-row>
          </el-col>
          <el-col v-if="is_login" class="login-bar" :span="5">

```

```
<div class="cart-ico">
    <b></b>
    
    <span>购物车</span>
</div>
<div class="study">学习中心</div>
<div class="member">
    <el-dropdown>
        <span class="el-dropdown-link">
            
        </span>
        <el-dropdown-menu slot="dropdown">
            <el-dropdown-item>我的账户 <i class="el-icon-arrow-right"></i></el-
dropdown-item>
            <el-dropdown-item>我的账户 <i class="el-icon-arrow-right"></i></el-
dropdown-item>
            <el-dropdown-item>我的账户 <i class="el-icon-arrow-right"></i></el-
dropdown-item>
            <el-dropdown-item>我的账户 <i class="el-icon-arrow-right"></i></el-
dropdown-item>
        </el-dropdown-menu>
    </el-dropdown>
</div>
</el-col>
<el-col v-else class="login-bar" :span="5">
    <div class="cart-ico">
        
        <span>购物车</span>
    </div>
    <span class="header-login">登录</span>
    &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&~
    <span class="header-register">注册</span>
</el-col>
</el-row>
</el-header>
</el-container>
</div>
</template>

<script>
export default {
  name:"Header",
  props:["current_page"],
  data(){
    return {
      is_login: false, /* 是否登录 */
    }
  }
}
</script>

<style scoped>
.header{
```

```

    width: 100%;
    height: 80px;
    background: #fff;
    position: fixed;
    left: 0;
    top: 0;
    margin: 0 auto;
    z-index: 7;
    text-overflow: ellipsis;
    white-space: nowrap;
    box-shadow: 0 0.5px 0.5px 0 #c9c9c9;
}
.el-container{
    width: 1200px;
    margin: 0 auto;
}
.el-header{
    padding: 0;
}
.el-row, .el-col{
    height: 80px;
}
.logo{
    width: 118px;
}
.logo, .nav{
    line-height: 80px;
}
.logo img{
    vertical-align: middle; /* 设置图片垂直居中 */
}
.nav{
    margin-left: 30px!important;
    margin-right: 0!important;
}
.nav .el-col .active{
    padding-bottom: 16px;
    padding-left: 5px;
    padding-right: 5px;
    border-bottom: 4px solid #ffc210;
}
.nav, .study{
    text-align: center;
    color: #4a4a4a;
}
.nav .el-col: hover, .nav .el-col a: hover, .study: hover, .study a: hover{
    color: #000000;
}
.login-bar{
    display: flex;
    align-items: center;
}
.cart-ico{

```

```

width: 88px;
height: 28px;
margin-right: 20px;
background: #f7f7f7;
border-radius: 17px;
display: flex;
align-items: center;
justify-content: space-around;
cursor: pointer;
position: relative;
font-size: 14px;
}
.cart-ico b{
width: 16px;
height: 16px;
line-height: 17px;
font-size: 12px;
color: #fff;
text-align: center;
background: #fa6240;
border-radius: 50%;
transform: scale(.8);
position: absolute;
left: 16px;
top: -1px;
}
.cart-ico img{
width: 15px;
height: auto;
margin-left: 6px;
}
.cart-ico span{
margin-right: 6px;
}
.study{
padding-left: 0;
font-family: PingFangSC-Regular;
letter-spacing: 0;
margin-right: 20px;
font-size: 15px;
cursor: pointer;
}
.member img{
width: 26px;
height: 26px;
border-radius: 50%;
display: inline-block;
cursor: pointer;
vertical-align: middle;
}
.member img:hover{
border: 1px solid rgb(255, 194, 16);
}

```

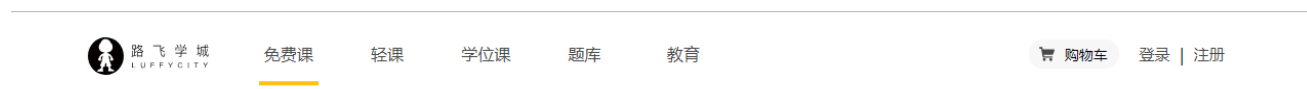
```

.el-dropdown-menu{
  left: 1130px!important;
  width: 180px;
  top: 64px!important;
}
.el-dropdown-menu i{
  float:right;
  line-height: 36px;
}
.header-login,.header-register{
  cursor: pointer;
  font-size: 15px;
  color: #4a4a4a;
}
.header-login:hover,.header-register:hover{
  color: #000000;
}
</style>

```

data数据中使用is_login控制显示登录和未登录状态的效果:

is_login:false



is_login: true



7. 购物车页面

7.1 创建购物车页面组件

在components/创建Cart .vue组件文件

```

<template>
  <div class="cart">
    购物车
  </div>
</template>

<script>
  export default {

```

```

    name: "Cart",
    data() {
      return {

      }
    }
  }
}
</script>

<style scoped>

</style>

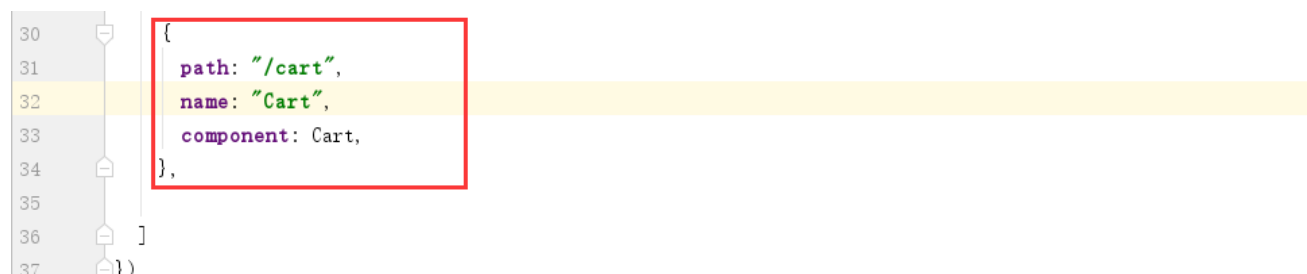
```

效果:



7.2 注册路由地址

routers/index.js代码:



7.3 在cart购物车组件中的头部和脚部引入

cart.vue代码:

```
<template>
  <div class="cart">
    <Header/>
    <div class="cart-info">

      </div>
    <Footer/>
  </div>
</template>

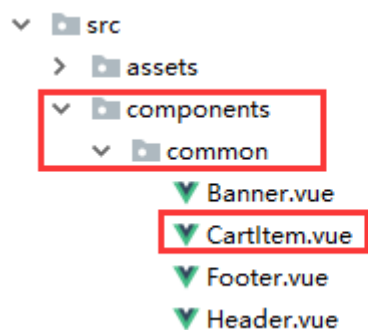
<script>
import Header from "../common/Header"
import Footer from "../common/Footer"
import CartItem from "../common/CartItem"
export default {
  name: "Cart",
  data() {
    return {

    }
  },
  components: {
    Header,
    Footer,
  }
}
</script>

<style scoped>
.cart{
  margin-top: 80px;
}
</style>
```

7.4 把购物车中信息作为子组件，引入到购物车页面中

在components/common目录下创建子组件文件CartItem.vue



CartItem.vue代码:

```

<template>
  <div class="cart-item">

  </div>
</template>

```

```

<script>
  export default {
    name: "CartItem",
    data() {
      return {

      }
    }
  }
</script>

```

```

<style scoped>

</style>

```

7.5 在Cart.vue购物车页面组件中引入CartItem商品信息组件

```

<template>
  <div class="cart">
    <Header/>
    <div class="cart-info">
      <h3 class="cart-top">我的购物车 <span>共1门课程</span></h3>
      <div class="cart-title">
        <el-row>
          <el-col :span="2">&nbsp;</el-col>
          <el-col :span="10">课程</el-col>
          <el-col :span="4">有效期</el-col>
          <el-col :span="4">单价</el-col>
          <el-col :span="4">操作</el-col>
        </el-row>
      </div>
      <CartItem/>
      <div class="calc">
        <el-row>
          <el-col :span="2">&nbsp;</el-col>
          <el-col :span="3">
            <el-checkbox label="全选" name="type"></el-checkbox></el-col>
            <el-col :span="2" class="del"><i class="el-icon-delete"></i>删除</el-col>
            <el-col :span="12" class="count">总计: ¥0.0</el-col>
            <el-col :span="3" class="cart-calc">去结算</el-col>
          </el-row>
        </div>
      </div>
    <Footer/>
  </div>

```

```

    </div>
</template>

<script>
  import Header from "../common/Header"
  import Footer from "../common/Footer"
  import CartItem from "../common/CartItem"
  export default {
    name: "Cart",
    data() {
      return {

      }
    },
    components: {
      Header,
      Footer,
      CartItem,
    }
  }
</script>

<style scoped>
.cart{
  margin-top: 80px;
}
.cart-info{
  overflow: hidden;
  width: 1200px;
  margin: auto;
}
.cart-top{
  font-size: 18px;
  color: #666;
  margin: 25px 0;
  font-weight: normal;
}
.cart-top span{
  font-size: 12px;
  color: #d0d0d0;
  display: inline-block;
}
.cart-title{
  background: #F7F7F7;
}
.cart-title .el-row, .cart-title .el-col{
  height: 80px;
  font-size: 14px;
  color: #333;
  line-height: 80px;
}
.calc .el-col{
  height: 80px;
}

```

```

    line-height: 80px;
  }
  .calc .el-row span{
    font-size: 18px!important;
  }
  .calc .el-row{
    font-size: 18px;
    color: #666;
    margin-bottom: 300px;
    margin-top: 50px;
    background: #F7F7F7;
  }
  .calc .del{

  }
  .calc .el-icon-delete{
    margin-right: 15px;
    font-size: 20px;
  }
  .calc .count{
    text-align: right;
    margin-right: 62px;
  }
  .calc .cart-calc{
    width: 159px;
    height: 80px;
    border: none;
    background: #ffc210;
    font-size: 18px;
    color: #fff;
    text-align: center;
    cursor: pointer;
  }
}
</style>

```

子组件CartItem.vue中编写组件代码,

```

<template>
  <div class="cart-item">
    <el-row>
      <el-col :span="2" class="checkbox"><el-checkbox label="" name="type"></el-
checkbox></el-col>
      <el-col :span="10" class="course-info">
        
        <span>Python开发21天入门</span>
      </el-col>
      <el-col :span="4">
        <el-select v-model="course_info.duration">
          <el-option v-for="item in options" :key="item.value"
:label="item.label" :value="item.value"></el-option>
        </el-select>
      </el-col>
    </el-row>
  </div>
</template>

```

```

        <el-col :span="4" class="course-price">¥9.0</el-col>
        <el-col :span="4" class="course-delete">删除</el-col>
    </el-row>
</div>
</template>

<script>
    export default {
        name: "CartItem",
        data() {
            return {
                options: [
                    {value: 30, label: "一个月有效"},
                    {value: 60, label: "二个月有效"},
                    {value: 90, label: "三个月有效"},
                    {value: -1, label: "永久有效"},
                ],
                course_info: {
                    id: 1,
                    title: "Python开发21天入门",
                    duration: 60,
                    src: "../../assets/course_1.jpeg",
                }
            }
        }
    }
</script>

<style scoped>
    .cart-item {
        height: 250px;
    }
    .cart-item .el-row {
        height: 100%;
    }
    .course-delete {
        font-size: 14px;
        color: #ffc210;
        cursor: pointer;
    }
    .el-checkbox, .el-select, .course-price, .course-delete {
        display: flex;
        align-items: center;
        justify-content: center;
        height: 100%;
    }
    .el-checkbox {
        padding-top: 55px;
    }
    .el-select {
        padding-top: 45px;
        width: 118px;
        height: 28px;
    }

```

```

    font-size: 12px;
    color: #666;
    line-height: 18px;
  }
  .course-info img{
    width: 175px;
    height: 115px;
    margin-right: 35px;
    vertical-align: middle;
  }
  .cart-item .el-col{
    padding: 67px 10px;
    vertical-align: middle!important;
  }
  .course-info{

  }
</style>

```

APP.vue中新增基本样式:

```

<style>
body,p{
  margin:0;
  padding:0;
}
a{
  color: #4a4a4a;
  text-decoration: none;
}
img{
  width: 100%;
}
.el-checkbox__inner{
  border: 1px solid #9b9b9b;
  border-radius: 0;
  width: 16px;
  height: 16px;
}
.el-checkbox__inner::after{
  height: 8px;
  width: 6px;
}
</style>

```

效果:

我的购物车 共1门课程

课程		有效期	单价	操作
<input type="checkbox"/>		Python开发21天入门	二个月有效	¥9.0 删除
<input type="checkbox"/> 全选		 删除		总计: ¥0.0 去结算

作业讲解

事件[Event]，就是DOM操作提供的功能。

DOM 操作中一共提供了三种事件操作级别给我们使用。

DOM0，使用基本的使用基本的事件绑定元素操作。

```
元素.onclick = function(){  
}
```

DOM2，使用事件的冒泡过程，对事件发生的元素进行捕获操作。

```
$(“ul”).on("click","li",function(){  
    // 处理li  
})
```

DOM3，js原生操作中会提供自定义事件给我们。

在用户操作数据/元素的，无非还是原来的js事件，但是我们通过 DOM3对这些事件，进行单独的自定义操作。例如修改事件名称，在别的元素中，判断新的事件的是否执行了。