

前言：能白嫖下次一定继续白嫖

宁可累死自己，也要卷死他人，卷死他人记住重复做几件事

1.打开网盘下载的资料开始学习，打开哔站开始看视频：

<https://space.bilibili.com/98307693> 陈天狼 综合全栈技术 微信：没有微信，嫖资料就行了，还想嫖我，哔站私信问我就行了

<https://space.bilibili.com/2060764481> 前端乐哥 前端殿堂级别西装技术男 微信：qingle101

能白嫖下次一定继续白嫖，千万不要点赞投币三连，顺带叫上，舍友，班级群，吹水群，三姑六姨群，一起白嫖他们

2.开始联系我们，然后让我拉你到进对应技术支持的群（进去就秃头那种），你自己能搞的定的话就别加我们了，求你少嫖一次

3.然后开始看一个视频，跟着做出来效果，然后在群、朋友圈、博客发布你的作品

4.重复第1，第3的动作，不断循环

注意：如果你真的遇到问题搞不定了，在点下面链接看吧，没问题别点，全是广告

【金山文档】技术支持社区（万一广告群），一对一指导（万一遇到噶腰子）

<https://kdocs.cn/l/ct1r2hsj8yQk>

#

Vue积分商城项目

项目参照地址：<http://kumanxuan1.f3322.net:8360/static/store-pc/index.html>

项目备份地址：<http://codesohigh.com/store-pc/home>

真实项目参考地址：<http://sc.wolfcode.cn/>

接口文档地址：<http://www.docway.net/project/1h9xcTeAZzV/share/1iUU09vKhMm>

axios官网网址：<http://www.axios-js.com/>

一、清空项目非必要文件

- `views` 下面的文件只保留 `Home.vue`，其余删除，删除 `components/HelloWorld.vue`，并且 `Home.vue` 中不再引入 `HelloWorld` 组件。
- 删除 `src/assets` 下的图片，换成我们的img文件夹。
- 将 `router/index.js` 中 `about` 的路由注释掉。
- 删除 `App.vue` 中的less。

二、样式初始化

开始书写结构样式，在App.vue组件中：

```
<div class="header"></div>

</script>
<style lang="less" scoped>
.header{
  height: 50px;
  background-color: #333;
}
</style>
```

发现页面会有自带的间距，这是浏览器本身的默认样式，所以我们需要进行样式初始化，清除浏览器默认样式。

安装初始化样式库reset-css：

```
npm i reset-css 或者 yarn add reset-css
```

安装成功后在main.js中引入即可：

```
import "reset-css"
```

三、网站结构布置

在App.vue中设置好头部，导航和尾部组件：

```
<template>
  <div id="app">
    <Tabbar></Tabbar>
    <Header></Header>
    <router-view/>
    <Footer></Footer>
  </div>
</template>
<script>
import Header from '@components/Header'
import Tabbar from '@components/Tabbar'
import Footer from '@components/Footer'

export default {
  components:{
    Header,Tabbar,Footer
  },
}
```

```
}  
</script>
```

在@/components目录下新建Header，Tabbar，Footer三个组件即可

四、网站数据请求模块

接口文档地址：<http://www.docway.net/project/1h9xcTeAZzV/share/1iUU09vKhMm>

接口地址：<http://192.168.113.249:8081/cms>

baseUrl 备用地址(不在广州校区内网的，请统一使用这个地址)：<http://kumanxuan1.f3322.net:8881/cms>

4.1、发起请求

作为一个网站前端，数据请求模块少不了。我们需要安装axios模块：

```
npm i axios
```

尝试在app.vue中做数据请求：

```
import axios from "axios"  
export default {  
  ...  
  created(){  
    axios.get("http://192.168.113.249:8081/cms/products/recommend")  
      .then(res=>{  
        console.log(res.data);  
      })  
      .catch(err=>{  
        console.log(err);  
      })  
  },  
}
```

4.2、代理配置

我们对 `vue.config.js` 进行配置：

```

module.exports = {
  devServer: {
    port: 8080,
    proxy: {
      '/api': {
        target: "http://192.168.113.249:8081/cms",
        pathRewrite: {
          '^/api': ''
        }
      }
    }
  }
}

```

由于配置文件修改了，这里一定要记得重新 `yarn serve` ！！

4.3、API与Request封装

在 `src` 下新建 `request` 目录，在 `request` 目录下新建 `request.js`

`request.js` 中：

```

import axios from "axios"

const instance = axios.create({
  baseURL: "http://192.168.113.249:8081/cms",
  timeout: 5000
})

instance.interceptors.request.use(config=>{
  console.log("每一次发起请求前，都会先执行这里的代码");
  console.log(config); //config本次请求的配置信息
  return config
},err=>{
  return Promise.reject(err)
})

instance.interceptors.response.use(res=>{
  console.log("每一次接收到响应，都会先执行这里的代码，再去执行成功的那个回调函数then");
  return res
},err=>{
  return Promise.reject(err)
})

export default instance

```

为了更好地管理我们的这些接口，我们把所有请求都抽取出来在一个`api.js`中

在 request 目录下新建 api.js, api.js 中:

```
import request from './request'

// 请求精品推荐数据
export const JingpinAPI = () => request.get('/products/recommend')
```

4.4、发起请求

App.vue 中:

```
import {JingpinAPI} from "@request/api"

created(){
  JingpinAPI()
    .then(res=>{
      if(res.errno == 0){
        console.log(res.data) // 成功拿到所有首页数据
      }
    })
}
```

五、头部组件

5.1、版心样式

在assets在新建css目录, 新建public.less文件

```
.wrap{
  width: 1200px;
  margin: 0 auto;
}
```

在main.js中全局引入

```
import "@assets/css/public.less"
```

5.2、头部组件布局

```
<template>
  <div class="header">
    <div class="wrap header-wrap">
      <div class="l">
        欢迎来到叩丁严选
      </div>
    </div>
  </div>
```

```

        <div class="r">
            <ul>
                <li class="avatar">
                    
                    用户名: --
                </li>
                <li>
                    我的鸡腿: --
                </li>
                <li>获取鸡腿</li>
                <li>叩丁狼官网</li>
                <li class="login-btn">登录</li>
            </ul>
        </div>
    </div>
</div>
</template>

<script>
export default {
    data () {
        return {

        }
    }
}
</script>

<style lang = "less" scoped>
    .header{
        height: 40px;
        background: #242B39;

        .header-wrap{
            color:#FFFEFE;
            height: 40px;
            font-size: 14px;
            /*      background-color: #fcf; */
            display: flex;
            justify-content: space-between;
            align-items: center;
            .r ul{
                display: flex;
                align-items: center;
                li{
                    margin-right: 20px;
                    cursor: pointer;

```

```

    }

    .avatar{
      display: flex;
      align-items: center;

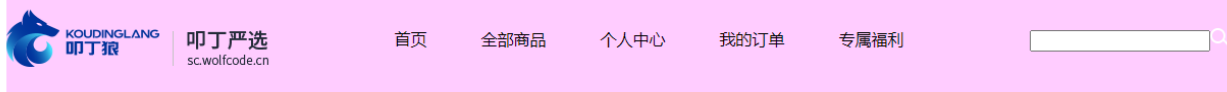
      img{
        border-radius: 50%;
        margin-right: 5px;
      }
    }

    .login-btn{
      width: 124px;
      height: 40px;
      text-align: center;
      line-height: 40px;
      background: #0A328E;
    }
  }
}
</style>

```

六、导航组件

6.1、基本布局



Nav.vue组件中：

```

<template>
  <div class="nav">
    <div class="wrap nav-wrap">
      <div class="l">
        <h1>
          
        </h1>
      </div>
      <div class="c">
        <ul>
          <li>首页</li>
          <li>全部商品</li>
          <li>个人中心</li>
          <li>我的订单</li>
          <li>专属福利</li>
        </ul>
      </div>
    </div>
  </div>
</template>

```

```

        </ul>
      </div>
      <div class="r">
        <input type="text">
        <span></span>
      </div>
    </div>
  </template>

  <script>
  export default {
    data () {
      return {

    }
  }
  }
  </script>

  <style lang = "less" scoped>
    .nav-wrap{
      height: 118px;
      background-color: #fcf;
      display: flex;
      justify-content: space-between;
      align-items: center;
      .c ul{
        width: 500px;
        display: flex;
        justify-content: space-between;
      }
    }
  </style>

```

6.2、搜索框布局



```

.r{
  display: flex;
  input{
    width: 214px;
    height: 40px;

```



```

        border: 1px solid #dedede;
        border-radius: 20px 0 0 20px;
        float: left;
        box-sizing: border-box;
        padding-left: 19px;
        outline-style: none;
    }
    .search-btn{
        width: 50px;
        height: 40px;
        background: #0A328E;
        border-radius: 0px 20px 20px 0px;
        text-align: center;
        line-height: 44px;
    }
}

```

七、路由配置及导航项当前样式

在router/index.js中配置重定向，及几个导航路由：

```

const routes = [
  {
    path: '/',
    redirect: '/home'
  },
  {
    path: '/home',
    name: 'Home',
    component: Home
  },
  {
    path: '/goods',
    name: 'Goods',
    component: () => import(/* webpackChunkName: "goods" */
'../views/Goods.vue')
  },
  {
    path: '/user',
    name: 'User',
    component: () => import(/* webpackChunkName: "user" */ '../views/User.vue')
  },
  {
    path: '/order',
    name: 'Order',
    component: () => import(/* webpackChunkName: "dingdan" */
'../views/order.vue')
  },

```

```

{
  path: '/free',
  name: 'Free',
  component: () => import(/* webpackChunkName: "free" */ '../views/Free.vue')
}
]

```

在View目录中新建对应组件。

处理导航项当前样式，在Nav.vue中

```

<div class="c">
  <ul>
    <li :class="$route.path===' /home'? 'active': ''">首页</li>
    <li :class="$route.path===' /goods'? 'active': ''">全部商品</li>
    <li :class="$route.path===' /user'? 'active': ''">个人中心</li>
    <li :class="$route.path===' /order'? 'active': ''">我的订单</li>
    <li :class="$route.path===' /free'? 'active': ''">专属福利</li>
  </ul>
</div>
<style>
...
.c ul{
  width: 500px;
  display: flex;
  justify-content: space-between;
  color: #242B39;
  font-size: 16px;
  font-family: SourceHanSansSC-Medium;
  font-weight: 500;
  .active{
    color: #0A328E;
  }
}
</style>

```

并且设置点击跳转路由:

```
<ul>
  <li @click="$router.push('/home') "
:click="$route.path=== '/home'? 'active': ''">首页</li>
  <li @click="$router.push('/goods') "
:click="$route.path=== '/goods'? 'active': ''">全部商品</li>
  <li @click="$router.push('/user') "
:click="$route.path=== '/user'? 'active': ''">个人中心</li>
  <li @click="$router.push('/order') "
:click="$route.path=== '/order'? 'active': ''">我的订单</li>
  <li @click="$router.push('/free') "
:click="$route.path=== '/free'? 'active': ''">专属福利</li>
</ul>
```