一、搭建环境+创建项目

1、下载node.js并安装（http://nodejs.cn）

2、下载好node之后，打开docs管理工具（开始右击-运行-cmd），先看看node安装成功了没有，输入 node -v ，回车，会输出node的版本号【安装node需要在网上查找教程，并且需要配置环境变量】

3、推荐使用淘宝npm镜像，使用淘宝的cnpm命令管理工具可以代替默认的npm管理工具，命令符执行：npm install -g cnpm --registry=https://registry.npm.taobao.org,验证是否安装成功，在命令输入npm -v，出来npm的版本号，及说明安装成功

4、安装vue 输入命令：npm install vue -g 。验证是否安装成功，在命令输入vue -v，出来vue的信息，及说明安装成功

5、vue安装成功之后，我们就可以全局使用vue-cli脚手架，输入命令：cnpm install --global vue-cli 回车；

6、创建项目（搭建完手脚架之后，我们要开始建一个新项目，建议尽量不要装在C盘，因为vue下载下来的文件比较大，如果要改盘的话，直接输入D：回车就可以直接改盘，或者提前创建好放项目的文件夹，在文件夹的路径上输入cmd运行docs，此后的操作就都是基于这个文件夹了。创建好项目后，docs操作具体某个项目，比如安装依赖或者运行项目时，同样在该项目文件夹路径上输入cmd后就可以直接针对该项目进行命令符操作了）

（1）vue-cli脚手架版本为3.0以下：

在命令输入vue init webpack my-project 回车，my-project是自己的文件夹的名字，是基于webpack的项目，输入之后就一直回车，直到出现是否要安装vue-router，

这个我们在项目要用到，所以就输入y 回车

（2）vue-cli脚手架版本为3.0以上：

方法A：在命令输入vue ui 回车，启动浏览器打开工程管理页面，直接手动创建项目

方法B：在命令输入vue create 项目名称 回车，根据提示输入命令即可成功创建项目

7、安装依赖（因为各个模板之间都是相互依赖的，所以现在我们要安装依赖）

输入命令：cnpm install

8、运行项目

方法A：使用intelliJ IDEA编辑器,将项目文件夹添加进编辑器后即可直接在编辑器上运行项目

方法B：打开docs（找到具体项目），输入npm run serve 回车 （脚手架3.0版运行方式-通过vue-ui创建的项目用此方法运行）

方法C：打开docs（找到具体项目），输入npm run dev 回车（脚手架2.0版运行方式-通过命令行创建的项目用此方法运行）

【补充：vue-cli脚手架version2和3的使用及更新的问题解决：

1. 首先卸载旧版本—npm  uninstall vue-cli –g
2. npm i @vue/cli –g（包名称由 vue-cli  改成了   @vue/cli）
3. 安装更新之后 ， 通过 vue -V  查看 vue-cli 安装的版本，如果版本没有发生变化，排除掉 npm 环境的问题，那么可能是因为缓存，这时，需要清理缓存。注意：使用管理员身份运行此命令清除缓存npm  cache  clear  --force
4. npm –V 查看版本号，应该已经变成3.0+版本了
5. @vue/cli（3.版本） 和 vue-cli（2.版本）的区别【文件结构以及配置文件有区别，另外创建项目和运行项目也有区别】
6. 安装@vue/cli 3的版本后还想使用vue-cli 2的版本？默认情况下，@vue/cli已经没有  vue  init   webpack  命令了，可通过全局安装一个桥接工具npm i -g @vue/cli-init即可使用   vue  init   webpack  项目名称          创建项目创建的依然是vue-cli  2.版本的 vue项目（包含build和config目录等）

】

9、rem-自适应（屏幕尺寸自适应匹配设置，px2rem-loader（Vue:将px转化为rem，适配移动端））

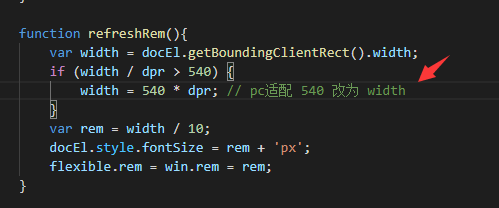
（1）下载lib-flexible（docs命令行：npm i lib-flexible --save）

（2）引入lib-flexible（在main.js中引入lib-flexible：import 'lib-flexible/flexible'）

（3）设置meta标签（index.html中通过meta标签，设置设备宽度以及缩放比例）

（4）安装px2rem-loader（docs命令行：npm install px2rem-loader）

（5）找到目录 ：node\_modules/lib-flexible/flexible.js ，修改flexible.js；



（6）配置px2rem-loader（在build文件中找到util.js，将px2rem-loader添加到cssLoaders中）

const cssLoader = {

loader: 'css-loader',

options: {

minimize: process.env.NODE\_ENV === 'production',

sourceMap: options.sourceMap

}

}

const px2remLoader = {

loader: 'px2rem-loader',

options: {

remUnit: 75,//这里是指设计稿宽度为750/10

remPrecision: 3//小数点后保留三位

}

}

同时，在generateLoaders方法中添加px2remLoader

function generateLoaders (loader, loaderOptions) {

const loaders = options.usePostCSS ? [cssLoader, postcssLoader, px2remLoader] : [cssLoader, px2remLoader]

**【注：vue3.0+步骤六不同：只需要在package.json内增加**

**"postcss": {**

**"plugins": {**

**"autoprefixer": {},**

**"postcss-px2rem": {**

**"remUnit": 130//这里是指设计稿宽度为1300/10**

**}**

**}**

**}】**

（6）修改flexible.js文件（在node\_module下找到lib-flexible文件中的flexible.js中得refreshRem方法）

function refreshRem() {

var width = docEl.getBoundingClientRect().width;

if (width / dpr > 540) {

width = width \* dpr;

}

var rem = width / 10;

docEl.style.fontSize = rem + 'px';

flexible.rem = win.rem = rem;

}

（7）当配置完之后，只需要重启下服务，就自动转化为rem了（项目样式中单位正常使用px即可）

10、vue2.0--axios的跨域问题

（1）安装axios——docs运行安装 npm install axios –save【3.0+运行：npm install axios -S】

（2）配置——在 src/main.js 中如下声明使用：

import axios from 'axios';Vue.prototype.$axios=axios;

（3）在 config/index.js 中的 的dev 添加以下代码，设置一下proxyTable

dev: {    //加入以下    dev: { proxyTable: { '/api': { target: 'http://172.17.0.211',//后端接口地址 //secure: true,//https时需要设置此项 changeOrigin: true,//是否允许跨越 pathRewrite: { //'^/api': '/', // 这种接口配置出来 http://XX.XX.XX.XX:8083/api/login '/api': ' ' //这种接口配置出来 http://XX.XX.XX.XX:8083/login }, //如果访问域名是http的，则需要加上如下代码【https不需要】 onProxyReq(proxyReq) { if (proxyReq.getHeader("origin")) { proxyReq.setHeader("origin","http://172.17.0.211") } } }},（4）在 config/dev.env.js 中设置以下代码：

module.exports = merge(prodEnv, {  NODE\_ENV: '"development"',  //开发环境  API\_HOST:"/api"})

（5）在 config/prod.env.js 中设置以下代码：

module.exports = {  NODE\_ENV: '"production"',//生产环境  API\_HOST:'"http://api.douban.com"'//接口域名

}

（6）main.js内添加axios.defaults.baseURL = process.env.API\_HOST（该名和4、5步内配置的名称保持一致）

（7） src/components内的vue 文件（实际调用接口）

this.$axios.post('/hlp-pms-app/rest/webServiceOpen/adapter', { 'ServiceData': { 'loginName': username, 'pswd': password }, 'ServiceType': 'login'}) .then(response => { console.log(response) if (response.data.code === 'S') { // 缓存token并且跳转页面 localStorage.setItem('token', response.data.data.token) this.$router.push({path: 'ParkList'}) } else if (response.data.code === 'F') { // 弹出登录失败提示 this.$message(response.data.msg) } }) .catch(err => { this.$message('登录失败，请稍后重试！')// 弹出接口调用失败提示 })

1. VUE3.0以上需自创建三个配置文件（在该文档同级的文件夹“vue3.0以上自创建文件”内提供），

【根据不用的环境定义baseurl：（获取当前访问url的域名作为接口请求域名）

let url = window.location.href.split("/");

let baseUrl = url[0]+"//"+url[2];

】

并在main.js内配置环境变量

// 判断环境

if(process.env.NODE\_ENV === "production"){

if(process.env.VUE\_APP\_FLAG === "pro"){

axios.defaults.baseURL = baseUrl;

}else{

axios.defaults.baseURL = baseUrl;

}

}else{

axios.defaults.baseURL = "/api";

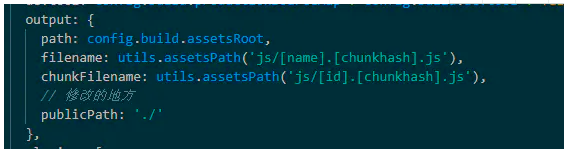
}

11、vue打包后index.html一片空白甚至报错

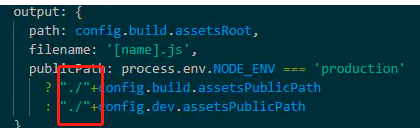


解决方法：

（1）在build文件里面找到webpack.prod.config.js，在output里面添加publicPath:'./';



（2）找到webpack.base.configjs



（3）如果router/index.js文件里面有mode: 'history’的话删去或者注释即可

（4）重新打包

（5）1-4步骤配置完后，访问URL会有“#”，想解决该问题：  
需要服务器端配置nginx，前端不执行步骤1-4，只需要在router/index.js内设置

mode: 'history’， base: '/rfs/'（base内的值为服务器端根目录，也就是存放打包后文件的上级目录至根目录）

12.axios的post和get请求

（1）get(headers和params放到同一个大括号内，可以放在params的前面，以下是有headers和无headers的两种示例)

**this**.$axios.get("/hlcloud-bms-app/cash/queryOpEntityBankList",{  
 params:{  
 opEntityid:**this**.fkOpEntityTenant  
 }  
}).then((res)=>{  
 console.log(res)  
 **if**(res.data.code=="S"){  
 **this**.companybanklist=res.data.data//获取公司账户列表  
 }**else**{  
 **this**.$message(res.data.msg)  
 }  
})  
 .catch(err=>{  
 **this**.$message('网络出错啦，请稍后重试')  
 console.log(err)  
 })



（2）post(headers要放在params的后面，不能和params在一个参数中，以下是有headers的示例)

**this**.$axios.post("/hlcloud-bms-app/cash/query", {  
 page:**this**.pagenum,  
 rows:**this**.handleSize,  
},{  
 headers: {  
 'content-type': 'application/x-www-form-urlencoded'  
 }  
}).then((res)=>{  
 console.log(res)  
 **if**(res.data.code=="S"){  
 **this**.recordstotal=res.data.data.records//获取总条数并赋值  
 // 获取列表数据并赋值  
 **this**.listtable=res.data.data.data;  
 }**else**{  
 **this**.$message(res.data.msg)  
 }  
})  
 .catch(err=>{  
 **this**.$message('网络出错啦，请稍后重试')  
 console.log(err)  
 })

学习/参考网址：

https://www.cnblogs.com/VoiceOfDreams/p/11052447.html

https://cli.vuejs.org/zh/config/#chainwebpack