npm install -g @vue/cli vue的脚手架3的安装方式

这个自动安装vue ui 可视化视图 可以看到vue版本的信息和一些配置的文件

vue create 项目的名称 vue cli3的初始化项目

npm install @vue/cli-init -g vue脚手架2版本的拉取

vue inint webpack 项目名称 vue cli2初始化的项目

vuex的安装方法

npm install vuex –save

npm init 是初始化一个文件 package.json 打包的意思 里面有安装好的项目信息

npm install 根据package.json里的信息来安装好对应的文件以供使用

补充知识 1. vue-router

npm install vue-router –save 安装方式 因为是运行时依赖的所以是 –save

vue ui 打开vue的可视化的界面

配置信息

1. vue build 基于两个结构 第一个建议初学者安装 第二个大神安装的

2. es（js）lint 这个插件就是帮助你检查你的js的代码是否规范

3. unit tests 单元测试 对某些模块单独的进行测试 一般公司用的少

4. e2e 端到端的测试 使用selenium来对浏览器进行自动化测试的框架 如一些几点按 钮的测试啊就1000万个按钮你都要手动点的吗？不可能吧，这个时候需要这个了

6. ESlint和在vue3中的 Linter是一个意思的 这个玩意的作用就是规范代码的不能有一点的错误，否则就拨错了

7. TypeScript 这个就是结合js和什么我忘了就是更加的完善代码，减少代码

MVVM:6 p55

model：模型，数据对象（data）

view：视图，模板页面

viemModel：视图模型（Vue的实例）

在vue的实例中存在的方法

created(){} 在创建完组件的时候开始回调（首先你这个方法要写到一个组件中的，个人理解是，当你这个方法中的组件使用时，此方法开始回调）

mounted(){} 在组件挂载到DOM中，执行此方法

updated(){} 在界面刷新的时候，执行此方法

生命周期：$destroy()调用此方法的时候 整个vue会结束程序 记得加括号

页面加载完成后执行： mounted（）{} 注意他是个函数

销毁前的回调函数 beforeDestroy(){} 当使用了$destroy此函数执c

在model中储存data的（数据）

view是DOM中有两个属性 一个是指令 另一个是{{ }} 绑定的数据

vue中的常用的指令： 都写在html中的

1. 强制数据绑定 v-bind

强制数据绑定v-bind 举例：在vue的data中添加一个img的属性

img:’一个图片的地址’ 正常的话使用<img src=”img”> 图片不会出来的但是那

<img v-bind:src=”img”>或<img :src=”img”> 就是加v-bind之后会强制的把img当成一个表达式来进行判断的，之前他会当成一个dom的元素

1. 绑定事件监听 v-on 后面带的都是一个函数|方法

<button v-on:click=”test”>qqq</button> 或者

<button @click=”test”>qqq</button> 如果想要传入参数的话就

<button @click=”test(‘参数’)”>qqq</button> 这时你需要在methods中添加形参

此时在vue中新创建一个方法 注意在事件后面是添加一个方法 比如test就是一个方法

事件一般都是一个方法

methods:{test(){alert(“qqq”)} }

1. 双向数据绑定 v-model

v-model 双向数据绑定，一般用于表单元素。就是在input中输入多少在body中就会显示多少。

使用方法在js中

New Vue({

el:”这里是id的值如#名字或class值如.名字”, el是element的缩写就是获取元素

el:’ #lei ’,

data：{ name：“”“， 这个data是数据的替换与存储

age：21，

”}

});

1. v-if=”true” 这个是删除元素在添加的 表达元素显示和隐藏
2. v-else=”false” 其实只写v-else就行 他的值跟随v-if变化的
3. v-show=”true||false” 这个是隐藏在显示的 比if用的灵活



1. v-for() 遍历设置的对象

v-for=”(元素值随便设置，index下标) in 对象或数组” 括号中的两个参数是可以随便设置的第一个参数是对象中的某个元素，第二个参数是下标 两个值可以随便的设置

上面说的是数组或对象中的对象 遍历数组还是有一个不一样的

遍历数组的时候 就是v-for=”(value,attribute) in 数组” 第一个参数是属性值

第二个参数是属性名

在原生的js中for in 数组中变量是下标 对象中变量是属性名

<div id="app">

<li v-for="(p,k) in persons" :key="index">{{p}}={{k}}={{p.name}},{{p.age}}</li>

</div>

<script>

var vm = new Vue({

el:'#app',

data:{

persons:[

{name:'Bom',age:11},

{name:'Rouse',age:32}

]

}

})

1. ref:指定唯一标识，vue对象通过$els属性访问这个元素对象
2. v-cloak:防止闪现，与css配合：【v-cloak】{display:none}

vue中的一些属性

1. 计算属性 computed

computed属性对象中定义计算属性的方法在也页面中使用{{方法名}}来显示计算的结果

在html中使用这个属性中定义的方法 页面一上来就会调用此方法的

使用方法 在input中使用v-model:”在computed中定义的方法的名称”

调用时间 1.页面加载后立即调用

2.当绑定的input中数据改变立即调用

3.相关的数据发生改变立即调用 比如在函数中定义一个return 在这里绑定data中的数据

并且在computed中有一个get（）和set（） 在getter中使用缓存数据存储数据

两个方法

在computed中单独设置一个属性赋值其对象，在对象中写入这两个方法在set方法的参数中有一个，这个是当前元素的值

get(){

return this.name +this.name1

},set(value){

const names = value.split('')

this.name = names[0]

this.name1 = names[1]

}

1. 监视属性 $watch（）或者watch

通过vm对象的$watch（）或者watch配置来监视指定的属性，当属性变化时，回调函数会自动调用，在函数内部进行计算 在watch中写入你想要监视的data的属性就行后面写上自定义的回调函数，而且回调函数中有两个形参new value，old value 一个是新的值一个是旧的值 $watch（）的使用方法在全局的

使用方法： new vue（）你赋值的那个变量（人家叫实例）.$watch（‘监视的data中的属性’，function（newvalue，oldvalue）{}） 当你监视的data中的数据改变的时候就会执行watch括号中的第二个参数回调函数他的参数有两个你可以随便的命名的一个就是那个更改的数据的新的值被赋值给第一个参数，另一个呐oldvalue举例就是更改数据的之前的数据赋值给了参数

vm.$watch('name',function (newvalue，oldvalue){

this.name1 = newvalue

})

绑定class 和style

元素中 class=“dom元素” :class=“这里就是变量” :class=” ’这里写就变成dom元素’ ”

:class=”{ }” :class=”[ ]” 必须使用强制数据v-bind: 的简写: 在class中可以传入数组和对象 :class=”{classA:isA,classB:isB}” 对象中都是变量 需要在data中去写入数据 在绑定对象时，对象中的属性就是原来的dom，但是他的属性值是一个变量，属性值还是原来的值不变

<p :class="{p:inA}">绑定class</p>

<p :style="{color: color1,background: back}">绑定style</p>

在时间的是处理中 vue中调用函数 在html中调用函数时传入一个$event也可以不写但是要在定义的实参之后 之前也可以 就是在事件的回调函数中存在一个自定义的形参event这个形参 可以用event.target指向调用函数的那个元素

在vue中停止事件的冒泡直接 @click.top=””

阻止事件的默认行为 @click.prevent 例子就是一个a标签去一个网站你可以用此阻值

指定某个按键 行为 @keyup.13=”” 就是表达你按键触发回车键时触发函数

1. mounted 承载的意思 相当于js中的window.onload

他是个函数 mounted（）{ 记得在函数中写箭头函数，因为箭头函数没有this，他会直接在外面找到vue}

vue的生命周期



vue router

前端路由：从后端的api接口中吧数据请求过来 ，通过js渲染

输入或者请求一个url地址，去打包的文件中寻找对象的url的组件，

后端路由：有后端吧页面给渲染好，直接发送给前端去展示

在网页的控制台中有一个 哈希值 location.hash =”” 这个存在于栈中的，在栈中啊值是先放后出的，后放的值是先出的。