1. 核心思想：Angular：MVC,模块化，双向数据绑定；React:独创的JSX语法，组件化；Vue:数据驱动，组件化，虚拟DOM
2. Vue事件处理器：

|  |  |  |  |
| --- | --- | --- | --- |
| 指令 | 含义 | 指令 | 含义 |
| v-on:click.stop | 表示阻止冒泡 | v-on:click.prevent | 阻止默认事件 |
| v-on:click.self | 只触发自身的事件 | v-on:click.once | 只执行一次 |

1. 前端路由

在单页面应用（SPA）中，看似有10个页面，但其实只有一个主页面，其他9个页面都不需要重新从服务器获取，用户体验极好。

* 动态路由

|  |  |  |
| --- | --- | --- |
| 模式 | 匹配路径 | $route.params |
| /user/:username | /user/evan | {username:’evan’} |
| /user/:username/post/:post\_id | /user/evan/post/123 | {username:’evan’,post\_id:123} |

* 嵌套路由

路由里面添加路由

* 编程路由
* 命名路由和命名视图

1. Vue-resource（不在推荐）和Axios（使用这个）

{

axios({  
 **url**:**"../package.json"**,  
 **method**:**"get"**,  
 **data**:{  
 **userId**:**"101"** },  
 **params**:{  
 **userId**:**"102"** },  
 **headers**:{  
 **token**:**"http-test"** }  
}).then(res=>{  
 **this**.**msg** = res.**data**;  
})

}

Axios这种方式的get请求通过params传参，而post请求是通过放在data里面。

拦截方式：

mounted: **function** () {  
 axios.interceptors.**request**.**use**(**function** (config) {  
 ***console***.log(**"request init."**);  
 **return** config;  
 })  
 axios.interceptors.**response**.**use**(**function** (response) {  
 ***console***.log(**"response init."**);  
 **return** response;  
 })  
},

1. ES6学习

* 扩展运算符…

动态rest传参

let sum3 = (...m)=>{

let total = 0;

for(var i of m){

total += i;

}

console.log(`total:${total}`)

};

sum3(4,8,9,10);

数组扩展

let arr1 = [1,3];let arr2 = [4,8];

console.log([…arr1,…arr2])

字符串扩展

let xy = [...'ES6'];

* Promise

基本用法：

**let** *testPromise*=()=>{  
 **return new *Promise***((resolve,reject)=>{  
 **let** test={  
 **userId**:**'123'** }  
 resolve(test);  
 })  
}  
*testPromise*().then((resolve)=>{  
 **if**(resolve.**userId**===**'123'**){  
 ***console***.log(**`success:**${resolve.**userId**}**`**)  
 }  
},(reject)=>{  
   
})

**all函数用法：**

***Promise***.all([*checkLogin*(),*getUserInfo*()]).then(([res1,res2])=>{  
 ***console***.log(**`result1:**${res1.**result**}**,result2:**${res2.**userId**}**`**)  
})

* Import和export

1. 如果我们用default，

export default {}

则import的时候

import 名称 from ‘文件路径’

1. 如果我们用

export let sum =(x,y)=>{

return x+y

}

则import的时候,必须用下面这种格式，

import {sum} from ‘文件路径’ 或者 import \* as util from ‘文件路径’

1. AMD、CMD、CommonJS和ES6
2. AMD（RequireJS）异步模块加载（依赖前置，提前执行）

define(['/package/lib'],function (lib) {

//做很多事情

return {

lib:lib

}

})

1. CMD(seaJS)同步模块加载（依赖就近，即用即拿）
2. CommonJS（module.exports）用在nodejs模块管理中
3. ES6（import/export）
4. vue-lazyload（图片懒加载）

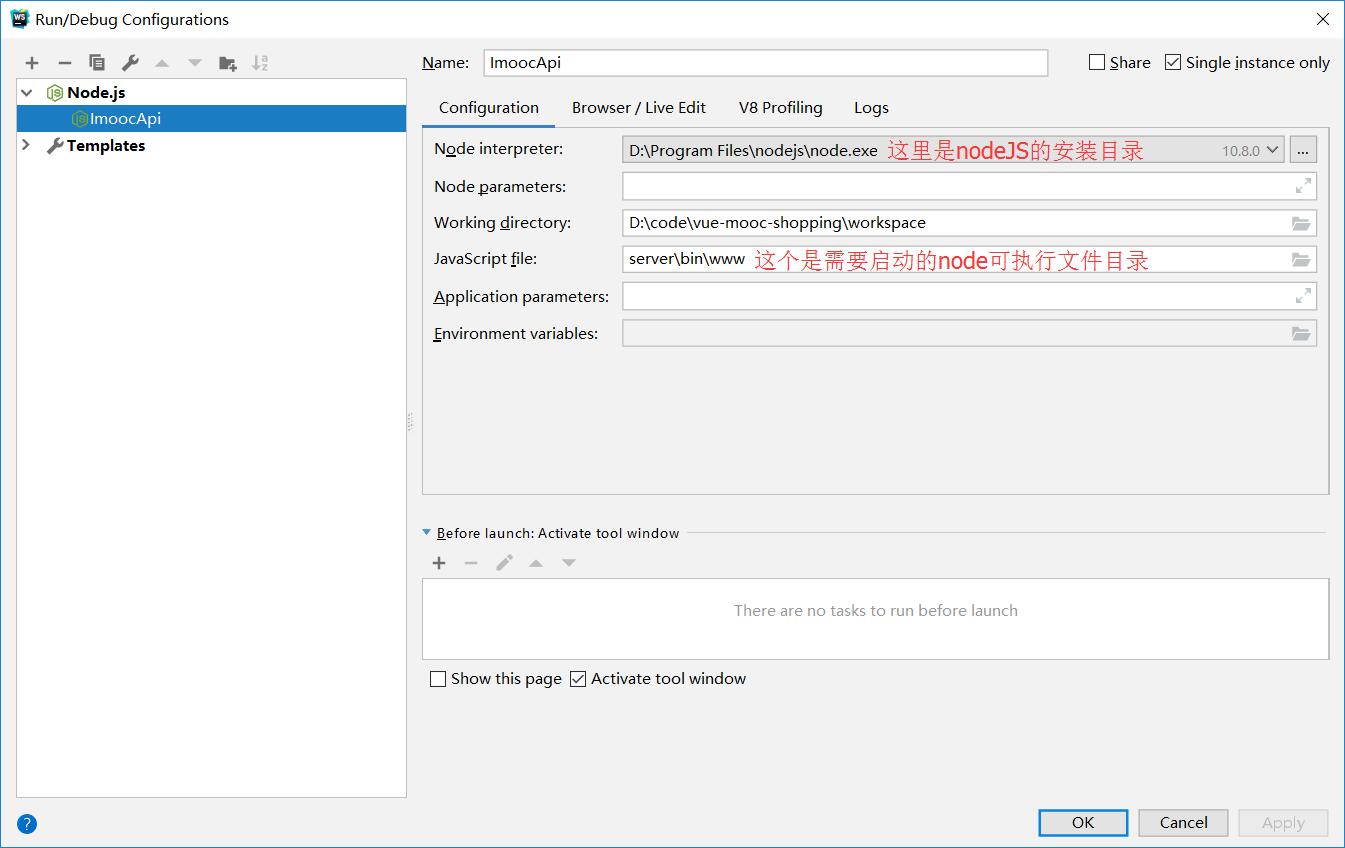
import Vue from 'vue'

import VueLazyLoadImg from 'vue-lazy-load-img'

Vue.use(VueLazyLoadImg)

1. 使用webstorm启动node后端服务的配置

选择菜单栏的Run，选择Edit Configrutions



1. 分页插件（滚动时加载）

vue-infinite-scroll

1. vuex(状态管理)



1. 登录密码输入完成后点击enter进行登录

**@keyup.enter="**login**"**

12、