复习：

见思维导图

今日目标：

(1)Vue.js实例、声明周期等小知识

(2)Vue.js中自定义组件 —— 重点&难点

1.如何在Vue.js中发起异步请求

(1)使用第三方工具库： jQuery.ajax( )

(2)使用官方插件：Vue-Resource

(3)使用第三方工具： Axios

var url = 'http://127.0.0.1/productlist.php?pno=2&pageSize=8';

axios.get(url).then(function(res){

console.log('异步请求成功');

console.log(res);

}).catch(function(err){

console.log('异步请求失败');

console.log(err);

})

|  |
| --- |
| Vue.js生态系统（Ecosystem）：  Vue.js、Axios、Vue-Router、Vuex、MintUI、ElementUI、Weex.... |

2.Vue实例可能出现成员

var vm = new Vue({

el: '#app',

data: { }, //数据

methods: { }， //方法，操作数据

watch: { }, //监听，监听数据的改变

computed: { },

props: { },

beforeCreate(){ },

created(){ },

beforeMount(){ },

mounted(){ },

beforeUpdate(){ },

updated(){ },

beforeDestroy(){ },

destroyed(){ }  
 })

consoe.log(vm); => vm.$el vm.$data vm.$watch

3.Vue.js实例的生命周期 —— 重点理论



Vue.js实例的生命周期分为四个阶段：

(1)创建(create)阶段

(2)挂载(mount)阶段

(3)更新(update)阶段

(4)销毁(destroy)阶段

为了监听四个阶段，Vue.js提供了八个钩子函数

beforeCreate(){ } 实例创建之前自动调用

created(){ } 实例创建完成自动调用

beforeMount(){ } 实例挂载到DOM树之前调用

mounted(){ } 实例挂载到DOM树之后调用

beforeUpdate(){ } 实例中模型数据发生改变需要更新DOM之前调用

updated(){ } 实例中模型数据发生改变需要更新DOM之后调用

beforeDestroy(){ } 实例被从DOM上销毁之前调用

destroyed(){ } 实例被从DOM上销毁之后调用

var vm = new Vue({ }); —— 此处的vm是**根组件(root)**

4.Vue.js中的自定义组件 —— 重点

Component：组件，是可复用的Vue实例，在视图中呈现为用户自定义的扩展标签。项目中，组件是页面中的一块区域，该区域中对应专有的模型数据、HTML/CSS结构。

|  |
| --- |
| **面向组件式的开发：**把大网页划分为若干组件组成的区域，每个区域都有专属性数据、HTML元素、CSS样式。  Component Tree |

自定义组件有两种方式：

**(1)全局组件**：可以在页面中任何一部分使用的组件

Vue.component('comp-name', {

template: ' ', //组件模板

data: function(){

return { } //专有的Model数据

},

props: ['属性1', '属性2', ...] //自定义属性(也是Model)

})

**(2)局部组件**：只能用于特定的父组件内的组件

Vue.component('comp-name', {

template: ' <child-comp-name/> ', //局部组件只能用在父组件内部

**components:** {

'child-comp-name': { //局部组件

template: '',

data: function(){return {}}

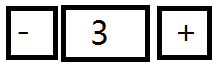
}

}

})

|  |
| --- |
| [Vue warn]: Error compiling template:  <h3>戴尔燃7000</h3>  <span class="text-info">￥6800.00</span>  - Component template should contain exactly one root element.  每个组件的模板内只允许有一个根元素 |
| [Vue warn]: The "data" option should be a function that returns a per-instance value in component definitions.  组件的data属性必须是一个函数，返回一个对象，其中包含组件中的专属数据——这一点与根组件不同：根组件只有一个实例，而每个自定义组件都可能被实例化多次，每个这样的组件实例都必须有自己专有的模型数据 |

练习：创建一个自定义组件：<buy-count></buy-count>

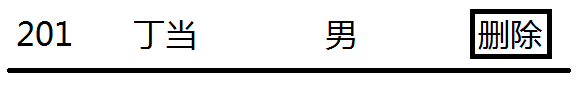


练习：创建一个自定义组件：<dd-book></dd-book>



练习：创建一个自定义组件：<tedu-emp></tedu-emp>，

每个该组件实例都可以声明eid、ename、sex三个自定义属性



页面中实例化4个该组件对象

课后练习：待办事项列表 Todo List

