正课:

1. DOM vs 函数库 vs 框架

2. 什么是Vue

3. 如何使用Vue

4. Vue的原理

5. 绑定语法

6. 指令

1. DOM vs 函数库 vs 框架

DOM: API繁琐

函数库: jQuery 对DOM的每个步骤的API进行一对一的简化，但并没有改变DOM做事的步骤和方法。

框架: 一个包含部分已经实现的功能的半成品项目，只是暂时缺少数据和个性化页面。

优点: 彻底简化了开发的步骤，且彻底避免了大量重复的劳动

缺点: 需要转变观念

2. 什么是Vue:

Vue是基于MVVM设计模式的渐进式的前端js框架

渐进式: 可以有选择的，逐渐使用框架中的功能

vs "全家桶": 必须完整使用所有功能，不能掺杂其它技术

前端js: Vue是纯前端的框架，无需node.js，就可独立运行。但是，需要node.js在运行之前将vue编译为浏览器认识的普通的HTML、css和js文件。

何时: 适合于以数据操作为主的项目

3. 如何使用Vue:

官网: https://cn.vuejs.org/

2种:

1. 下载并引入vue.js文件

最新版本: 2.5.x

开发版: 未压缩，带错误提示

生产版: 压缩，去掉错误提示

2. 安装脚手架工具

4. Vue的原理:

旧式的前端划分:

html: 定义网页静态内容

css: 定义网页静态样式

js: 为网页添加动态行为

MVVM设计模式:

新的前端划分:

1. 将一个.html文件，拆成多个组件文件: .vue/.component

2. 每个组件由细分为3部分:

1. 界面(View): HTML和CSS

2. 数据模型(Model): 在内存中自己创建的或从服务端请求来的保存页面需要的所有数据的对象或数组: ajax中的res

3. 控制器(ViewModel/controller): 自动将数据模型中的数据绑定/同步到页面需要的位置——不再需要程序员反复手工查找并修改元素。

控制器的原理: 2大子系统:

1. 响应系统: 监视模型数据的变化，并通知框架修改页面的机制。

其实是将数据模型中每个属性变为一个访问器属性

每次修改模型中的数据时，都会自动触发set

set会发出通知: "xx变量发生改变"

2. 虚拟DOM树: 扫描原始DOM而产生的一棵仅包含可能变化的元素的新的简版DOM树

优点:

1. 仅包含可能变化的元素的可能变化的个别属性——简单——遍历快

2. 修改时，可仅找到发生变化的元素，仅修改发生变化的元素——效率高

vue对象创建时，首次扫描并创建虚拟DOM树

响应系统发出通知时: 虚拟DOM树收到通知，并查找要修改的个别元素，实际修改DOM树。

5. 绑定语法:

什么是: 在页面中使用特殊的符号{{}}指定所需的变量名

学名: interpolation 插值语法

何时: 只要页面的某个位置需要动态生成值时

如何:

{{模型变量名}}

{{表达式}}

表达式包括: 算术计算，关系运算/逻辑运算，方法调用，访问数组元素，访问对象的属性，三目运算

强调: 不能写程序结构:

if else if else

for while do while

6. 指令(directive):

什么是: 为HTML提供新功能的特殊属性

为什么: {{}}只能绑定元素的内容(innerHTML)，不能绑定属性/事件或执行程序结构

何时: 不能使用{{}}绑定的位置，又想动态生成值，都要用各种指令来增强

如何:

1. v-bind: 专门用于动态绑定属性值

何时: 只要一个属性的值需要动态生成时

如何: 完整: <ANY ~~v-bind~~:属性="表达式"

可省略v-bind

2. v-for: 专门根据一个数组中的元素，动态反复生成多段相同的页面内容

何时: 只要根据一个数组中的元素，动态绑定页面内容时

如何: <ANY v-for="模型变量 of 数组类型模型变量"

根据of后数组中的元素个数，反复生成多个<ANY>元素

of前的模型变量是临时的，获得数组中的一个当前元素

of前的模型变量在<ANY内部可用于绑定！

3. 控制元素的显示隐藏:

v-if: 专门根据一个条件表达式来决定当前元素是否显示

只有条件表达式为true时，当前元素才显示

何时: 只要根据一个条件控制元素的显示隐藏时

如何:

<ANY v-if="条件表达式1"

<ANY v-else-if="条件表达式2"

<ANY v-else-if="..."

<ANY v-else//不用写条件

强调: v-if和v-else-if和v-else之间不能插入其它元素

v-show:专门根据一个条件表达式来决定当前元素是否显示

只有条件表达式为true时，当前元素才显示

采用display:block/none的方式显示隐藏

何时: 如果仅控制一个元素的显示和隐藏时，首选v-show

vs v-if: 采用添加删除元素的方式，控制显示隐藏

何时: 多个元素选其一显示时，首选v-if v-else-if v-else

4. 绑定事件:

界面View中:

<ANY v-on:事件名="处理函数()"

简写: v-on:可替换为@

new Vue({})中:

new Vue({

el:"选择器",

data:{ 模型变量 },

methods:{//专门保存控制器所需所有自定义方法的属性

处理函数(){//处理函数:function(){

//this->当前Vue对象->vm

this.模型变量 //访问模型变量:

}

}

})

问题1: 阻止默认行为:

解决: 2步:

1. 界面: 当事件发生时，自动调用js语句，并传入$event作为参数

<ANY @click="处理函数($event)"

$event就是DOM中的e，是vue封装的事件对象

2. new Vue()中:

methods:{

//定义形参，准备接事件对象

处理函数(e){

e.preventDefault();

}

}

简写: 事件修饰符，对DOM中事件API的简写:

2个:

阻止默认行为: @click.prevent="处理函数()"

取消冒泡: @click.stop="处理函数()" =>e.stopPropagation()