1. 基础

(1)Vue是MVVM框架, 数据的双向绑定

M -- Model(模型, 后端传递的数据) --> data

V -- View(视图, 所看到的页面) -->html

VM -- ViewModel(视图模型, 链接M和V的桥梁) --> new Vue

(2)实例参数(new Vue)

## el: “id” 实例所控制的html区域

## data: {} 存放的数据( 子组件必须是一个方法 return {} )

## methods: {} 存放方法

(3)模板语法

{{ }}

1. 指令

(1)v-text：展示其对应的文本

(2)v-html：把值中的标签渲染出来

(3)v-on：事件绑定

### 事件修饰符：

+ .stop 阻止冒泡

+ .prevent 阻止默认事件

+ .capture 添加事件侦听器时使用事件捕获模式

+ .self 只当事件在该元素本身（比如不是子元素）触发时触发回调

+ .once 事件只触发一次

(4)v-bind：适用于绑定行内属性

## v-bind:style

## v-bind:class

(5)v-for：循环显示元素的;可以循环数组, 也可以循环对象、数字、字符串

(注意必须绑定绑定key值)

##实例

<div v-for="(val, key, i) in userInfo">{{val}} --- {{key}} --- {{i}}</div>

(6)v-model：双向绑定

(7)v-show=‘布尔’：布尔为 true 显示，false 隐藏。

(8)v-if=‘布尔’：布尔为 true 显示，false 隐藏。 (惰性)

##v-show和v-if总结

一般来说，v-if 有更高的切换消耗而 v-show 有更高的初始渲染消耗。因此，如果需要频繁切换 v-show 较好，如果在运行时条件不大可能改变 v-if 较好。

(7)v-once：对应的标签只渲染一次

(8)v-cloak：防止页面加载时出现 vuejs 的变量名指令，在项目在开发中，经常会遇到当数据加载时vue的一些标签就会闪现，然后等数据加载完后消失，这时候就需要用到 v-cloak 来解决。