



Vue.js详解

Lesson 3

Vue数据绑定



授课老师：爱米

模板的说明 ——

- Vue.js 的模板是基于 DOM 实现的。
所有的 Vue.js 模板都是可解析的有效的 HTML，且通过一些特殊的特性做了增强。
Vue 模板因而从根本上不同于基于字符串的模板。

Vue.js数据绑定

- ☐ 1.插入值
- ☐ 2.绑定表达式
- ☐ 3.指令
- ☐ 4.缩写

插入值 ——

- ☐ 1.文本
- ☐ 2.原始的HTML
- ☐ 3.HTML特性

文本 ——

- 数据绑定最基础的形式是文本插值，使用 “Mustache” 语法（双花括号）：
- `Message: {{ msg }}`
- Mustache 标签会被相应数据对象的 msg 属性的值替换。每当这个属性变化时它也会更新。

文本 ——

- 当然，我们也可以只处理单次插值，今后的数据变化就不会再引起插值更新了：

- `This will never change: {{* msg }}`

原始的 HTML

- 双 **Mustache** 标签将数据解析为纯文本而不是 **HTML**。为了输出真的 **HTML** 字符串，需要用三 **Mustache** 标签：
- ```
<div>{{{ raw_html }}}</div>
```
- 内容以 **HTML** 字符串插入——数据绑定将被忽略。如果需要复用模板片断，应当使用 **partials**。

# HTML 特性

- Mustache 标签也可以用在 HTML 特性 (Attributes) 内:

- ```
<div id="item-{{ id }}"></div>
```

- 在 **Vue.js** 指令和特殊特性内不能用插值。
不必担心, 如果 Mustache 标签用错了地方 Vue.js 会给出警告。

绑定表达式 ——

☐ 1.JavaScript表达式

☐ 2.过滤器

JS表达式

- **Vue.js** 在数据绑定内支持全功能的 JavaScript 表达式：

- `{{ number + 1 }}`

- `{{ ok ? 'YES' : 'NO' }}`

- `{{ message.split("").reverse().join("") }}`

JS表达式

- 表达式将在所属的 **Vue** 实例的作用域内计算。每个绑定只能包含单个表达式。

- `<!-- 这是一个语句，不是一个表达式： -->`
`{{ var a = 1 }}`

`<!-- 流程控制也不可以，可改用三元表达式 -->`
`{{ if (ok) { return message } }}`

过滤器

- Vue.js 允许在表达式后添加可选的“过滤器 (Filter) ”。
- `{{ message | capitalize }}`
- 管道语法不是 JavaScript 语法，因此不能在表达式内使用过滤器，只能添加到表达式的后面。

过滤器 ——

○ 过滤器可以串联：

○ `{{ message | filterA | filterB }}`

过滤器 ——

○ 过滤器也可以接受参数：

○ `{{ message | filterA 'arg1' arg2 }}`

指令

○ 指令 (Directives) 是特殊的带有前缀 v- 的特性。

○ `<p v-if="greeting">Hello!</p>`

指令 ——

- 1.参数
- 2.修饰符

参数 ——

○ 有些指令可以在其名称后面带一个“参数”，中间放一个冒号隔开。

○ `<a v-bind:href="url">`

○ `<a v-on:click="doSomething">`

修饰符 ——

- 修饰符 (**Modifiers**) 是以半角句号开始的特殊后缀，用于表示指令应当以特殊方式绑定。

- `<a v-bind:href.literal="/a/b/c">`

缩写

- v- 前缀是一种标识模板中特定的 **Vue** 特性的视觉暗示。
- 在构建单页应用时，**Vue.js** 会管理所有的模板，此时 v- 前缀也没那么重要。
- 因此**Vue.js**为两个最常用的指令 **v-bind** 和 **v-on** 提供特别的缩写。

缩写 —



```
<!-- 完整语法 -->  
<a v-bind:href="url"></a>  
<!-- 缩写 -->  
<a ;href="url"></a>
```



```
<!-- 完整语法 -->  
<a v-on:click="doSomething"></a>  
<!-- 缩写 -->  
<a @click="doSomething"></a>
```

THANKS FOR WATCH