1. Vue2

1.模板语法

1. V-html:用于输出html代码
2. V-bind:v-bind:class="{ 'active': isActive }"
3. V-model:在input输入框中，实现双向数据绑定（input、select、textarea、checkbox、radio）
4. V-if:来决定是否插入此元素（true/false）
5. V-on:用来监听dom事件
6. 条件语句
   1. V-if：决定是否插入该元素
   2. V-else：用此指令给v-if添加一个else代码块
   3. V-else-if：用作v-if的else-if代码块
   4. V-show：是否显示该元素（通过样式的改变）
   5. v-if通过控制删除vue的虚拟dom上对应的节点，然后间接的控制删除真实dom树上的节点。 v-show是通过style样式 display：none来隐藏元素 v-if适合在通过数据控制元素显示隐藏的时候。 v-show适合在频繁的切换状态的时候。
7. 循环语句
   1. v-for：循环元素（类似于foreach）--key为变量，value为值，index为索引.
8. 计算属性
   1. 关键词：**computed（和data同级）**
   2. **关键词：Methods（和data同级）**computed 是基于它的依赖缓存，只有相关依赖发生改变时才会重新取值。而使用 methods ，在重新渲染的时候，函数总会重新调用执行。
   3. Computed-》属性-》get、set（读取 写入（赋值））
9. 监听属性
   1. Watch

vm.$watch('kilometers', function (newValue, oldValue) )

1. 样式绑定：
   1. <div v-bind:style="[baseStyles, overridingStyles]">菜鸟教程</div>
   2. new Vue({
   3. el: '#app',
   4. data: {
   5. baseStyles: {
   6. color: 'green',
   7. fontSize: '30px'
   8. },
   9. overridingStyles: {
   10. 'font-weight': 'bold'
   11. }
   12. }
   13. })
2. 事件处理器：
   1. V-on时事件修饰符

<!-- 阻止单击事件冒泡 --><a v-on:click.stop="doThis"></a><!-- 提交事件不再重载页面 --><form v-on:submit.prevent="onSubmit"></form><!-- 修饰符可以串联 --><a v-on:click.stop.prevent="doThat"></a><!-- 只有修饰符 --><form v-on:submit.prevent></form><!-- 添加事件侦听器时使用事件捕获模式 --><div v-on:click.capture="doThis">...</div><!-- 只当事件在该元素本身（而不是子元素）触发时触发回调 --><div v-on:click.self="doThat">...</div>

<!-- click 事件只能点击一次，2.1.4版本新增 --><a v-on:click.once="doThis"></a>

* .stop - 阻止冒泡
* .prevent - 阻止默认事件
* .capture - 阻止捕获
* .self - 只监听触发该元素的事件
* .once - 只触发一次
* .left - 左键事件
* .right - 右键事件
* .middle - 中间滚轮事件
* 按键修饰符.enter
* .tab
* .delete (捕获 "删除" 和 "退格" 键)
* .esc
* .space
* .up
* .down
* .left
* .right
* .ctrl
* .alt
* .shift
* .meta

实例

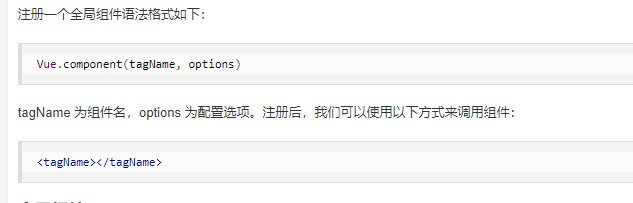
<p><!-- Alt + C --><input @keyup.alt.67="clear"><!-- Ctrl + Click --><div @click.ctrl="doSomething">Do something</div>

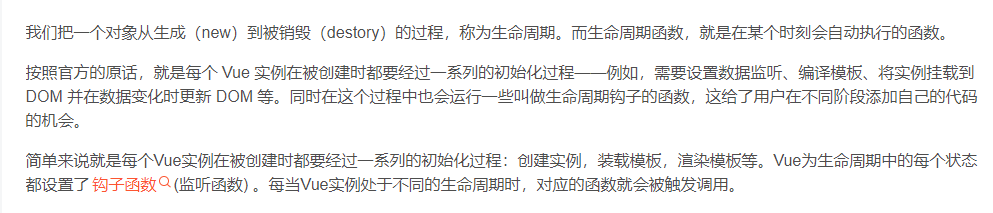
1. 表单
   1. v-mode 双向的数据绑定
   2. {{}}

<input v-model.trim="msg">去掉input值的首尾空格

<input v-model.number="age" type="number">将输入的值自动转换为number

<input v-model.lazy="msg" >在默认情况下， v-model 在 input 事件中同步输入框的值与数据，但你可以添加一个修饰符 lazy ，从而转变为在 change 事件中同步：

1. vue.js组件
   1. 
   2. 全局组件
   3. 局部组件
   4. Prop
   5. 动态prop
   6. prop 是单向绑定的：当父组件的属性变化时，将传导给子组件，但是不会反过来。
   7. Prop验证
2. 自定义事件
   1. 
3. 自定义指令
   1. Vue.directive
   2. 钩子函数



* 1. 

1. 过度&动画
   1. 
2. 混入
3. Ajax（axios）
4. Ajax(vue-resource)
5. 响应接口
6. 实例
7. 路由
8. Vue3