

《Vue基础》第四单元

方法定义和事件处理



方法定义

- 通过 **methods** 来定义
- methods 将被混入到 Vue 实例中，
可以通过实例访问(调用)
- 可以在插值或 v- 指令表达式中使用
- 方法中的 this 被绑定为 Vue 实例

方法定义

```
methods: {  
  getTime() {  
    const date = new Date();  
    return `${date.getHours()}:${date.getMinutes()}:${date.getSeconds()}`;  
  },  
  setTime() {  
    setInterval(() => {  
      this.time = this.getTime();  
    }, 1000);  
  }  
}
```

`vm.setTime();`

课堂案例：时钟

事件处理

- 用 **v-on** 指令监听 DOM 事件
- 可以把 JS 代码写在 v-on 指令中
- 也可以在 JS 语句中调用方法
- 通常是写一个要调用的方法名称
- 默认参数是 `$event`

事件处理

- click / focus / blur
- touchstart / touchmove / touchend
- mouseenter / mouseleave
- mouseover / mouseout
- keydown / keypress / keyup
-

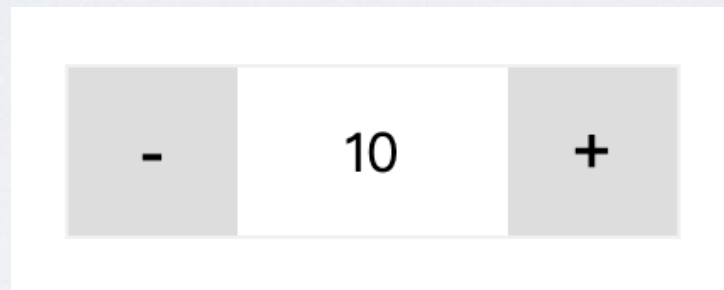
事件处理

- 可以把 JS 代码写在 v-on 指令中
- 也可以在 JS 语句中调用方法

```
<button v-on:click="num--">-</button>  
<div>{{ num }}</div>  
<button v-on:click="num++">+</button>
```


案例

课堂案例：购物车商品数量



- 案例地址 demo.jxsjs.com/step-input-1.html

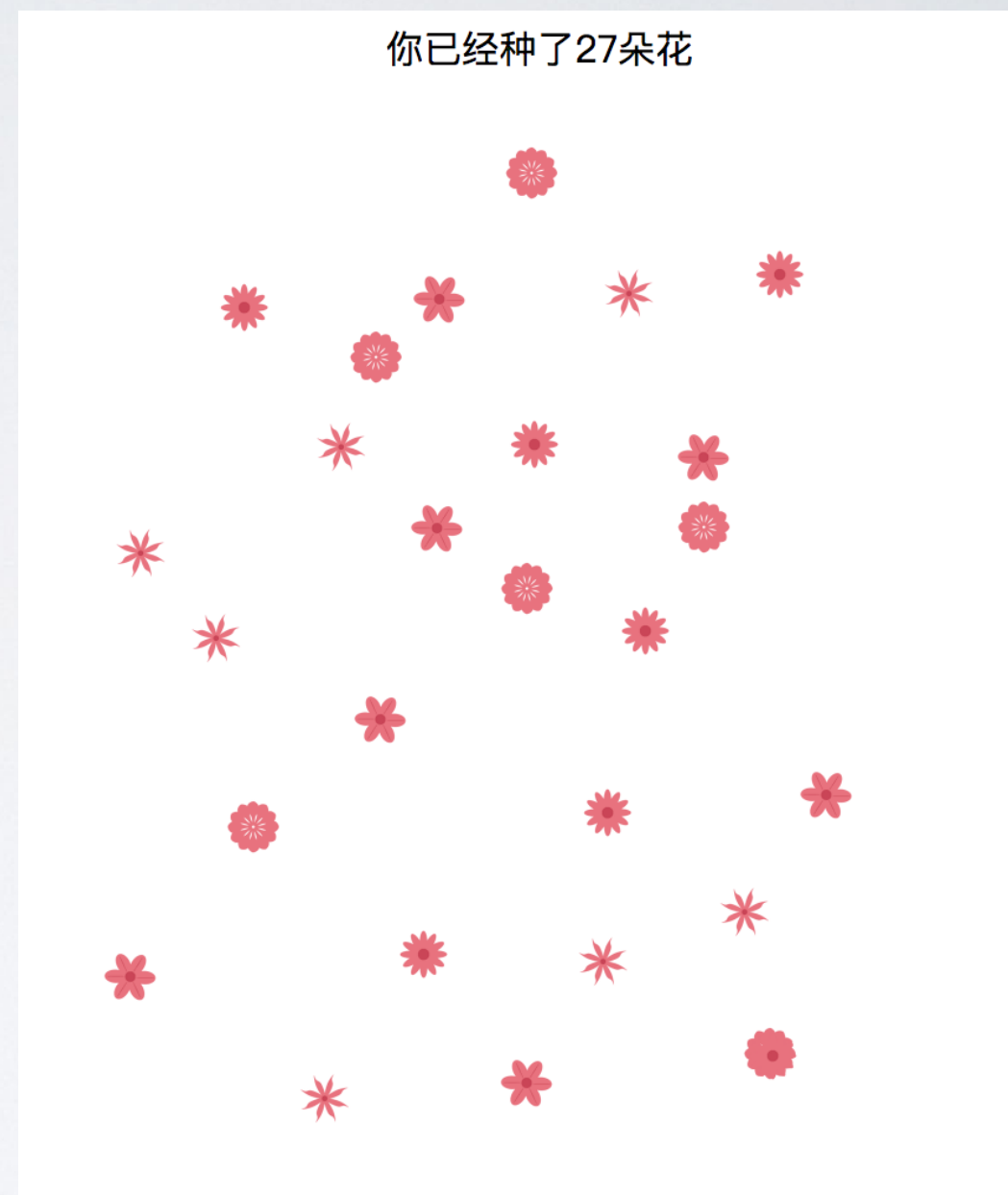
事件处理

- 通常是写一个要调用的方法名称
- 默认参数是 **\$event** , 原始 DOM 事件

```
<div class="farm" v-on:click="addFlower">
  <div
    v-for="(item, index) in flowerList"
    :key="index"
    :style="item"
    :class="['flower', 'flower-' + (index % 4 + 1)]"
  ></div>
  <div class="text">你已经种了{{ flowerList.length }}朵花</div>
</div>
```

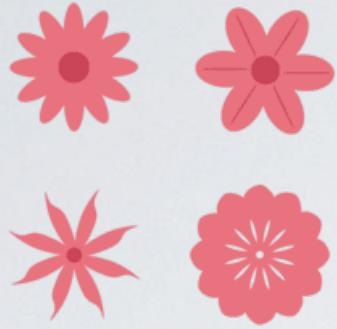

案例

课堂案例：种花



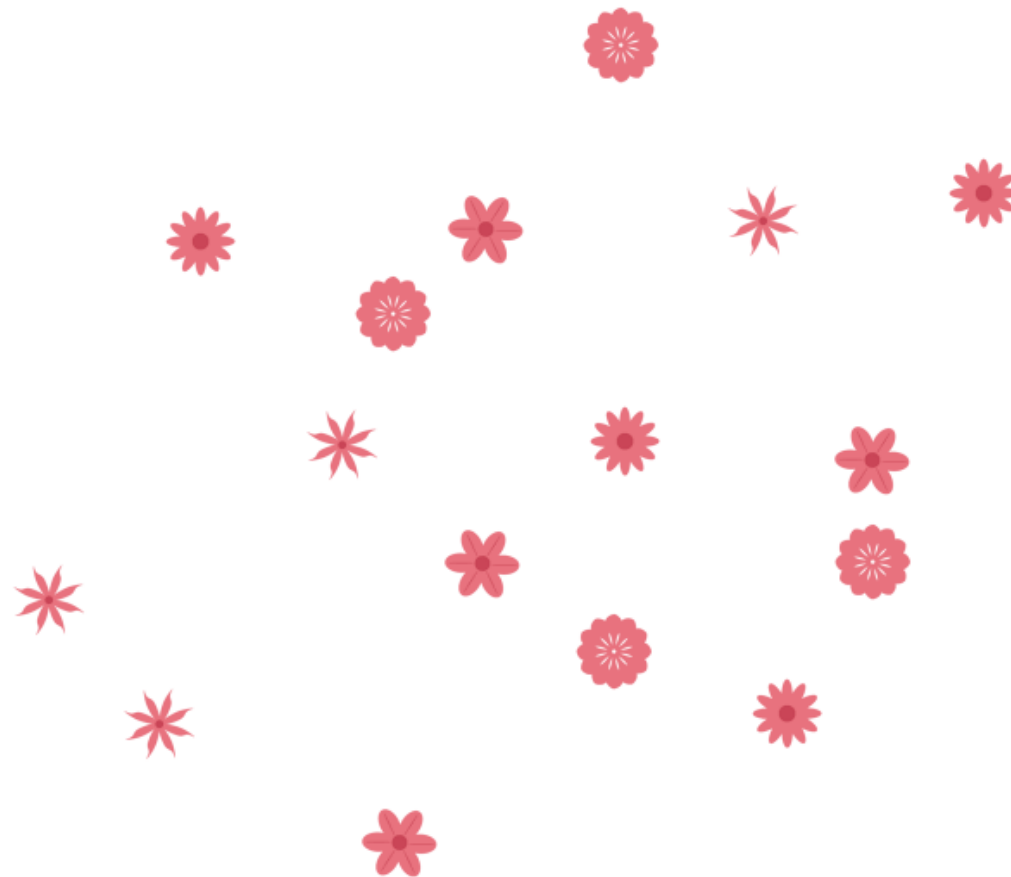
- 案例地址 demo.jxsjs.com/flower.html

素材地址



<http://jxsjs.com/img/flower.png>

你已经种了27朵花



事件修饰符

- 由点开头的指令后缀
- .stop `event.stopPropagation()`
- .prevent `event.preventDefault()`
- .self `event.target` 是当前元素自身
- .once 只触发一次

按键修饰符

- 监听键盘事件时，检查具体的按键
- 最常用的是 `v-on:keyup`
- `.enter` 回车
- `.space` 空格
- `.up .down .left .right` 方向键

缩写

- v-on:事件类型 → @事件类型

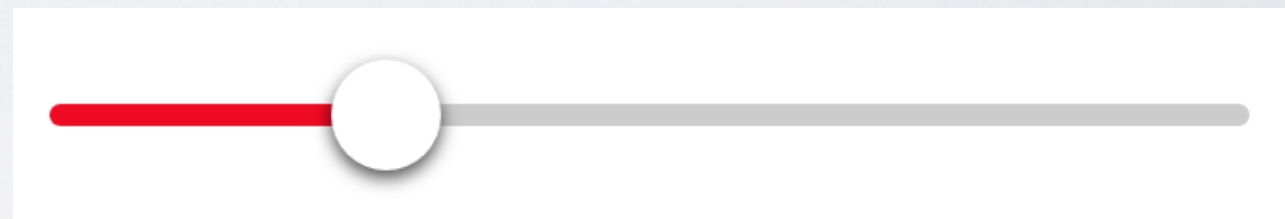
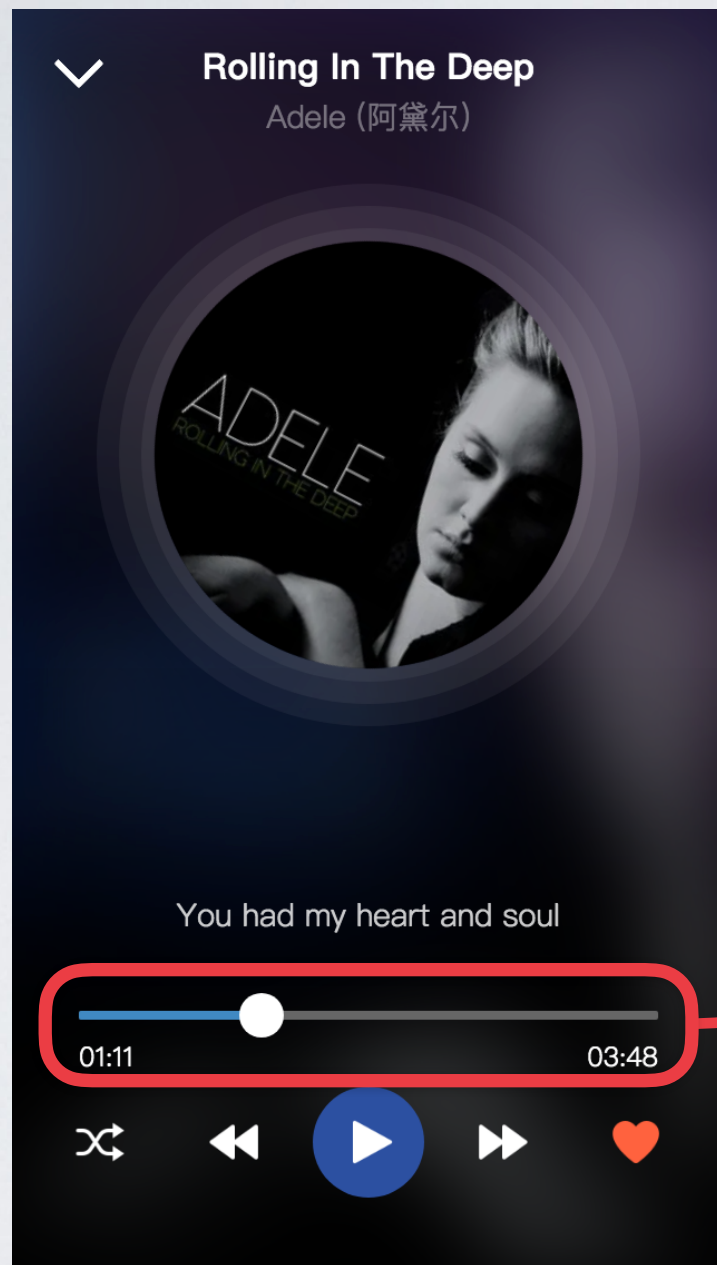
```
· <!-- 完整语法 -->  
· <a v-on:click="fn">...</a>  
· <!-- 缩写 -->  
· <a @click="fn">...</a>
```



回忆：插值绑定如何缩写？

案例

课堂案例：移动端音量控制滑块



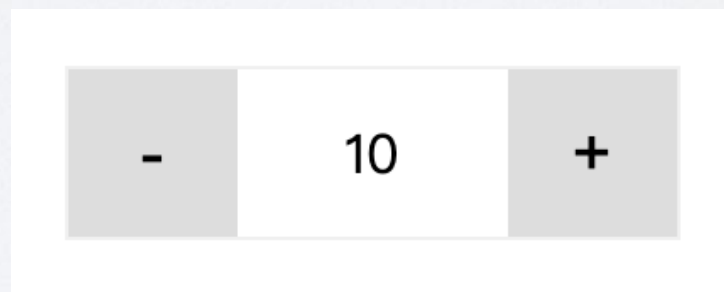
- 案例地址 demo.jxsjs.com/slider.html

总结

- methods
- this
- v-on (缩写)
- \$event
- 修饰符

课后作业

- 购物车商品数量，定义方法，增加功能
- 中间使用 input 输入
- 键盘回车时校验数字格式，非法报错，合格则提交



A UI component for quantity input. It consists of three parts: a gray square button with a minus sign (-) on the left, a white input field with the number 10 in the center, and a gray square button with a plus sign (+) on the right. All three parts are contained within a single white rectangular box.