

Vue 入门

目录

目标任务	2
相关知识点	2
思路分析	2
编码步骤	3
● 实现点击按钮时, 改变 input 内的 msg 内容为' andy'	3
● 实现练习: 使用 method, computed, watch, 当输入 firstname, lastname 时, 计算出 fullname	
4	
回马枪总结	10

目标任务

- 能够使用`{{}}`, `v-bind`, `v-model` 显示数据
- 能够使用 `v-on` 或`@`绑定事件
- 能够使用 `methods` 方法处理程序
- 能够使用 `computed`, `watch` 侦听属性变化
- 能够使用 `v-if`, `v-else-if`, `v-else`
- 能够使用 `v-show`
- 能够使用 `v-for`
- 能够定义和使用组件
- 能够使用组件的 `slot`
- 能够使用组件传参

相关知识点

- `{{}}`, `v-bind` 或:单向绑定, `v-mode` 双向绑定
- `Method`, `computed`, `watch`
- `V-if`, `v-show`, `v-else-if`, `v-else`
- `V-for`
- 组件的定义与使用, `slot`, `props`

思路分析

N/A

编码步骤

- 实现点击按钮时，改变 input 内的 msg 内容为 'andy'

1. 代码

```
App.vue
1  <template>
2    <div id="app">
3      
4      <h1>{{ msg }}</h1>
5      <!-- <input v-bind:value="msg" />
6      <input v-model="msg" />
7      <button @click="changeMsg">click me</button>
8    </div>
9  </template>
10
11
12  <script>
13  export default {
14    name: 'app',
15    data () {
16      return {
17        msg: 'Welcome to Your Vue.js App'
18      }
19    },
20    methods: {
21      changeMsg() {
22        this.msg = 'andy'
23      }
24    }
25  }
26  </script>
```

2. 运行效果




Welcome to Your Vue.js App

Welcome to Your Vue.js App



- 实现练习：使用 method, computed, watch, 当输入 firstname, lastname 时，计算出 fullname



目录

- 1 数据绑定 (v-bind, v-model)
- 2 事件与方法 (methods, v-on)
- 3 计算属性 (computed)
- 4 侦听属性 (watch)

练习：使用method, computed, watch, 当输入firstname, lastname时，计算出fullname
如：Andy Zheng

1. 代码

```

App.vue
8   <button @click="changeMsg">click me</button>
9   <hr />
10  firstname: <input v-model="firstname" /><br />
11  lastname: <input v-model="lastname" /><br />
12  fullname: {{firstname}}&nbsp;{{lastname}}<br />
13  fullname methods: {{getFullName()}}<br />
14  fullname computed: {{fullname}}<br />
15  fullname watch: {{fullnameWatch}}<br />
16
17 </div>
18 </template>
19
20 <script>
21 export default {
22   name: 'app',
23   data () {
24     return {
25       msg: 'Welcome to Your Vue.js App',
26       firstname: 'Andy',
27       lastname: 'Zheng',
28       fullnameWatch: 'Andy Zheng'
29     }
30   },
31   methods: {
32     changeMsg() {
33       this.msg = 'andy'
34     },
35     getFullName() {
36       return this.firstname + ' ' + this.lastname
37     }
38   },
39   computed: {
40     fullname: function() {
41       return this.firstname + ' ' + this.lastname
42     }
43   },
44   watch: {
45     firstname: function(val) {
46       this.fullnameWatch = val + ' ' + this.lastname
47     },
48     lastname: function(val) {
49       this.fullnameWatch = this.firstname + ' ' + val
50     }
51   }
52 }

```

2. 运行效果



Welcome to Your Vue.js App

Welcome to Your Vue.js App

firstname:
lastname:
fullname: Andy Zheng
fullname methods: Andy Zheng
fullname computed: Andy Zheng
fullname watch: Andy Zheng



Welcome to Your Vue.js App

Welcome to Your Vue.js App

firstname:
lastname:
fullname: AndyZ Zhenga
fullname methods: AndyZ Zhenga
fullname computed: AndyZ Zhenga
fullname watch: AndyZ Zhenga

- 实现以下练习



1 V-if, v-else-if, v-else

2 V-show

3 V-for

练习：输入name，通过后面的“添加”按钮把每次输入的name显示在下方。
当前行下标用3取余
为0时，显示背景颜色为红；
为1时，显示背景颜色为绿；
为2时，显示背景颜色为蓝；

1. 代码

```
App.vue
5  fullname watch: {{fullnameWatch}}<br />
6  <hr />
7
8  <input v-model="name" /><button @click="addUser">添加</button><br />
9  <ol>
10   <li v-for="(item, index) in userList" :key="index">
11     <span v-if="index % 3 == 0" class="red">{{item}} - {{index}}</span>
12     <span v-else-if="index % 3 == 1" class="green">{{item}} - {{index}}</span>
13     <span v-else class="blue">{{item}} - {{index}}</span>
14   </li>
15 </ol>
16 </div>
17 </template>
18
19 <script>
20 export default {
21   name: 'app',
22   data () {
23     return {
24       msg: 'Welcome to Your Vue.js App',
25       firstname: 'Andy',
26       lastname: 'Zheng',
27       fullnameWatch: 'Andy Zheng',
28       name: '',
29       userList: []
30     }
31   },
32   methods: {
33     addUser() {
34       this.userList.push(this.name)
35       this.name = ''
36     },
37     changeMsg() {
```

```

<style lang="scss">
.red {
  background-color: red;
}
.green {
  background-color: green;
}
.blue {
  background-color: blue;
}
</style>

```

2. 运行效果

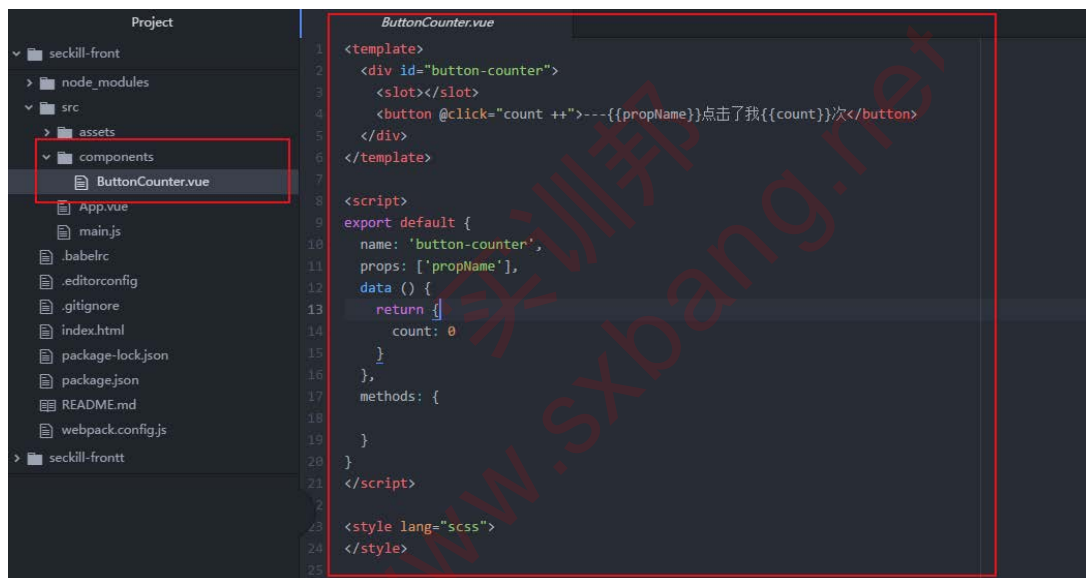


- 实现以下练习



3. 代码

ButtonCounter.vue 代码

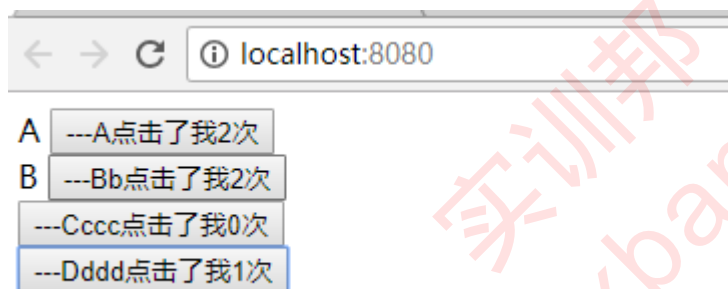


```

App.vue
1 <template>
2   <div id="app">
3     <button-counter :prop-name="a">
4       <span>A</span>
5     </button-counter>
6     <button-counter prop-name="Bb"><span>B</span></button-counter>
7     <button-counter prop-name="Cccc"></button-counter>
8     <button-counter prop-name="Dddd"></button-counter>
9   </div>
10 </template>
11
12 <script>
13 import ButtonCounter from './components/ButtonCounter'
14 export default {
15   name: 'app',
16   components: {
17     ButtonCounter
18   },
19   data () {
20     return {
21       a: 'A'
22     }
23   },

```

4. 运行效果



回马枪总结

- 1, 注意 v-if, v-show 的区别, v-show 只是样式隐藏
- 2, 注意 method, computed, watch 的区别, computed 会缓存
- 3, 组件传参需要在使用上定义参数名和值, 被使用方要使用 props 声名

