

Curso Técnico Superior Profissional em Tecnologias e Programação de Sistemas de Informação

Unidade Curricular: Desenvolvimento Web – Front-End

1.º Ano/2.º Semestre

Docente: Marco Miguel Olival Olim

Data 08/03/2019

ESTE EXERCÍCIO ABORDA OS CONCEITOS PRINCIPAIS DA FRAMEWORK DE FRONT-END VUEJS

Na continuidade da ficha anterior, vamos assinalar quais as tarefas já realizadas e as que são prioritárias. Para o efeito, sugere-se que seja efetuado um refactoring, alterando os elementos do Array para objetos, acrescentando-lhe mais duas chaves por tarefa – prioritária e realizada

```
22  el: '#app',
23  data: {
24    novaTarefa: '',
25    tarefas: [{
26      nome: 'Exercício de HTML e CSS',
27      prioritario: false,
28      realizada: false
29    }, {
30      nome: 'Exercício de JavaScript',
31      prioritario: false,
32      realizada: false
33    }, {
34      nome: 'Exercício de VueJS',
35      prioritario: false,
36      realizada: false
37    }, {
38      nome: 'Exercício de Angular',
39      prioritario: false,
40      realizada: false
41    }, ]
42  },
43  methods: {
```

No entanto, no browser a informação surge desta forma:

Lista de tarefas

- { "nome": "Exercício de HTML e CSS", "prioritario": false, "realizada": false }
- { "nome": "Exercício de JavaScript", "prioritario": false, "realizada": false }
- { "nome": "Exercício de VueJS", "prioritario": false, "realizada": false }
- { "nome": "Exercício de Angular", "prioritario": false, "realizada": false }

Cofinanciado por:

Para voltar ao formato de apresentação anterior basta indicar qual a chave pretendida:

```
15     <ul>
16     |     <li v-for="item in tarefas">{{ item.nome }}</li>
17     </ul>
18 </div>
```

Ao introduzirmos novos elementos deixa de ser renderizado o nome porque este novo elemento ainda não tem as novas chaves

← → ↻ ⓘ 127.0.0.1:5500/vue-1.html

Lista de Tarefas

- Exercício de HTML e CSS
- Exercício de JavaScript
- Exercício de VueJS
- Exercício de Angular
-

Adequamos então a função `insereTarefa`

```
43     methods: {
44         insereTarefa: function() {
45             if (!!this.novaTarefa) {
46                 this.tarefas.push({
47                     nome: this.novaTarefa,
48                     prioritario: false,
49                     realizada: false
50                 })
51                 this.novaTarefa = ''
52             }
53         }
54     }
55 }
```

O objetivo agora é modificar dinamicamente o estilo dos elementos da lista. Para tal, criamos as classes a serem aplicadas numa tag `style`

```
56 </script>
57 <style>
58     .realizada {
59         text-decoration: line-through;
60         color: grey;
61     }
62
63     .prioritario {
64         color: red;
65     }
66 </style>
67 </body>
```

Cofinanciado por:



Há duas sintaxes disponíveis para aplicar classes dinamicamente – Array e Objeto. A sintaxe de objeto é a mais simples e intuitiva, consistindo numa chave/valor (em que a chave é a classe a aplicar)

```
<li v-for="item in tarefas" :class="{realizada: item.realizada}">{{ item.nome }}</li>
```

A sintaxe por Array recorre a uma expressão ternária, tendo como primeiro argumento a condição a avaliar nos itens das tarefas:

```
<li v-for="item in tarefas" :class="[item.realizada ? 'realizada' : '']">{{ item.nome }}</li>
```

Podemos adicionar múltiplas classes em ambas as sintaxes, separando os elementos com vírgula

```
:class="{realizada: item.realizada, prioritario:item.prioritario}"
```

Vamos tirar partido da reatividade e alterar o estado da realização da tarefa simplesmente ao clicar sobre o item pretendido

```
15 <ul>
16   <li v-for="item in tarefas"
17     :class="{realizada: item.realizada, prioritario:item.prioritario}"
18     @click="realizaTarefa(item)">
19     {{ item.nome }}
20   </li>
21 </ul>
```

Esta nova função altera o valor atual da propriedade realizada do respetivo item tarefa para o seu contrário

```
47 methods: {
48   realizaTarefa: function(item) {
49     item.realizada = !item.realizada
50   },
51   insereTarefa: function() {
```

Pode agora alterar a realização de cada tarefa dinamicamente

← → ↻ ⓘ 127.0.0.1:5500/vue-1.html

Lista de Tarefas

- Exercício de HTML e CSS
- Exercício de JavaScript
- Exercício de VueJS
- Exercício de Angular

Cofinanciado por:

