

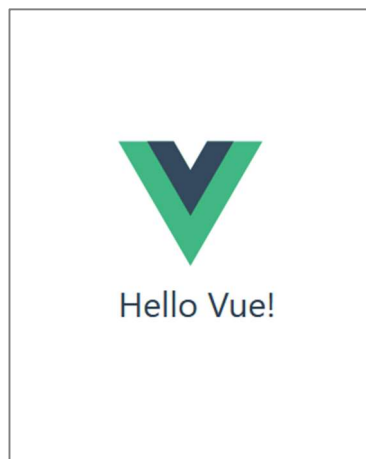
ALP

FICHA DE EXERCÍCIOS  
ATIVIDADE LETIVA

Programação Web I
Ficha 01 - Introdução ao Vue.js

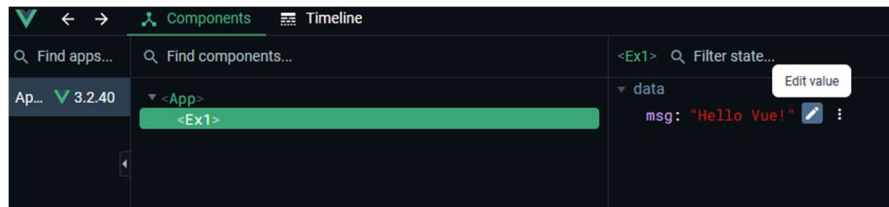
Abra o VSC e crie um projeto Vue chamado **VueIntro** através da linha de comando.

### Exercício 01



- 1) Crie um componente chamado **Ex1.vue** na pasta **components** do projeto Vue
- 2) Adicione a propriedade **msg** com o valor **"Hello Vue!"** na função **data** do componente criado
- 3) Exiba no template respetivo a propriedade **msg** dentro de uma tag **<h1>**
- 4) Altere o componente pai **App.vue** de forma que este importe o novo componente e o invoque. Para tal inclua **import Ex1 from './components/Ex1.vue'** no script do **App.vue**

- 5) Remova as importações dos outros componentes e a referência aos componentes no template
- 6) Adicione a referência ao componente **Ex1** através da inclusão de **<Ex1/>** no template
- 7) Verifique o resultado no browser.
- 8) Em caso de sucesso, abra o DevTools e altere a propriedade **msg** para outro valor e verifique se é feita a atualização do valor na janela do browser



## Exercício 02

- 1) Crie um novo componente chamado **Ex2.vue** na pasta **components**
- 2) Adicione no template do novo componente o seguinte HTML:

```
<div id="intro">
  <h2>First name: {{firstName}}</h2>
  <h2>Last name: {{lastName}}</h2>
  <h2>Age: {{age}}</h2>
  <h2>Full name: {{fullName()}}</h2>
</div>
```

- 3) Tendo em conta o conteúdo do template anterior, altere a secção **<script>** do componente de forma a dar resposta ao solicitado. Tenha em atenção que o método **fullName** devolve a junção dos valores das propriedades **firstName** e **lastName** com um espaço a separar.
- 4) Altere novamente o componente **App.vue** de forma a importar e invocar o novo componente (pode comentar/remover o anterior)
- 5) Altere o apelido da pessoa no DevTools e verifique o resultado

## Exercício 03

- 1) Crie um novo componente chamado **Ex3.vue** na pasta **components**
- 2) Adicione no template do novo componente o seguinte HTML:

```
<h2>Peso: {{weight}} kgs</h2>
```

- 3) Defina o valor **72** na propriedade **weight** na função data do componente.
- 4) Adicione todas as funções do ciclo de vida de uma instância/componente Vue e, em cada uma delas, inclua uma instrução **console.log(['nome\_evento'])**
- 5) Faça reload à página e verifique o resultado

- 6) Altere o valor da propriedade **weight** (via DevTools) e verifique o resultado

#### Exercício 04

- 1) Crie um novo componente chamado **Ex4.vue** na pasta **components**
- 2) Adicione na instância do novo componente o seguinte objeto:

```
productName: "Dell XPS"
productDesc: "Processador i7<br>Disco SSD 256GB<br> Monitor Samsung 21'"
productImage: "https://tinyurl.com/2p9dnx68"
```
- 3) Atualize o **App.vue**
- 4) Altere o template do componente de forma a renderizar os dados da instância do componente. Tenha em atenção o tipo de interpolações que usa
- 5) Adicione uma propriedade **productPrice** à função **data** com o valor **1200** e adapte o template de forma a renderizar esta nova informação
- 6) Aplique uma diretiva de forma que o valor desta propriedade apareça apenas se for inferior a **1000**. Dica: use a diretiva **v-if**
- 7) Altere o preço do produto (via DevTools) para 500 e verifique o resultado
- 8) Crie um botão que ao ser clicado reponha o preço original. Dica: deve usar a diretiva **v-on**
- 9) Adicione uma propriedade **productAvailability** à função **data** com o valor **true** e adapte o elemento HTML de forma a renderizar esta nova informação com:
  - a. texto **DISPONÍVEL** caso o valor seja **true**
  - b. texto **NÃO DISPONÍVEL** caso o valor seja **false**
- 10) Crie um botão que permita vender o produto e verifique o resultado final