

Deixando tudo mais alegre com efeitos

Transcrição

Vue não faz mágicas no que diz respeito à animações. É o desenvolvedor que deve criar os estilos que devem ser aplicados no momento correto para que os efeitos aconteçam. O que o Vue faz é indicar a hora exata na qual nossas classes devem ser aplicadas.

No entanto, o Vue precisa delimitar a área do template no qual fará essas indicações. Como queremos realizar uma transição no conteúdo do nosso `Painel`, vamos envolver nossa `div` com a classe `painel-conteudo` pelo componente `transition`. Obrigatoriamente precisamos escolher um `name` para ele. No caso, usaremos `painel-fade`:

```
<!-- alurapic/src/components/shared/painel/Painel.vue -->

<template>

  <div class="painel">

    <h2 class="painel-titulo" @dblclick="visivel = !visivel">{{ titulo }}</h2>
    <transition name="painel-fade">
      <div class="painel-conteudo" v-show="visivel">
        <slot></slot>
      </div>
    </transition>

  </div>

</template>

<script>
// código omitido
</script>

<style>
/* código omitido */
</style>
```

Em primeiro lugar, `transition` só pode ser usado com um elemento, se tentarmos colocar mais um elemento como filho do nosso wrapper receberemos um erro.

Se clicarmos em nossa `Painel`, nada ainda acontece. Precisamos criar o estilo para `painel-fade`. O problema é que para realizarmos uma transição, precisamos aplicar estilos diferentes, por exemplo, quando o elemento está sendo exibido ou quando ele está sendo escondido.

É por isso que o componente `transition` adiciona dinamicamente nos seus elementos filhos as seguintes classes dinamicamente:

```
painel-fade-enter // antes do elemento ser incluído ou removido, o estado atual
painel-fade-enter-active // quando o elemento está sendo incluído
painel-fade-leave-active // quando o elemento está sendo removido
```

Sendo assim, podemos fazer o seguinte:

```
<!-- alurapic/src/componets/shared/painel/Painel.vue -->

<template>

<div class="painel">

  <h2 class="painel-titulo" @dblclick="visivel = !visivel">{{ titulo }}</h2>
  <transition name="painel-fade">
    <div class="painel-conteudo" v-show="visivel">
      <slot></slot>
    </div>
  </transition>

</div>

</template>

<script>
// código omitido
</script>

<style>
/* código anterior omitido */

.painel-fade-enter, .painel-fade-leave-active {
  opacity: 0
}

</style>
```

Por fim, usamos os outros dois estilos para aplicar a transição:

```
<!-- alurapic/src/componets/shared/painel/Painel.vue -->

<template>

<div class="painel">

  <h2 class="painel-titulo" @dblclick="visivel = !visivel">{{ titulo }}</h2>
  <transition name="painel-fade">
    <div class="painel-conteudo" v-show="visivel">
      <slot></slot>
    </div>
  </transition>

</div>

</template>

<script>
// código omitido
</script>
```

```
<style>
/* código anterior omitido */

.painel-fade-enter, .painel-fade-leave-active {
  opacity: 0
}

.painel-fade-enter-active, .painel-fade-leave-active {
  transition: opacity .4s
}
</style>
```

Façamos um teste agora. Quando ocultamos e exibimos no elemento um fade é realizado.