

Módulo 1: Introdução ao Vue.js

- diretiva v-bind, atribuindo valor às tags html

A diretiva **v-bind** no Vue.js é utilizada para fazer a ligação (binding) dinâmica de valores a atributos de tags HTML. Isso significa que, em vez de você definir manualmente um valor fixo para um atributo de uma tag HTML, como **src** de uma imagem, **href** de um link, ou até mesmo classes e estilos, você pode utilizar o **v-bind** para definir esse valor dinamicamente, com base em dados reativos dentro do componente Vue.

Como funciona:

- A sintaxe básica é: **v-bind:atributo="expressão"**.
- **atributo** é o nome do atributo HTML que você quer vincular ao valor dinâmico.
- **expressão** é o valor dinâmico que você deseja atribuir ao atributo. Essa expressão pode ser uma propriedade, uma variável, ou até mesmo uma função definida no componente Vue.

Analisando o exemplo prático a seguir:

Estrutura geral do HTML:

```
<!DOCTYPE html>
<html lang="pt-br">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Curso Vue js - diretiva v-bind, atribuindo valor às tags html</title>
  <style>
    .red{
      color: ■ red;
    }
    .green{
      color: ■ green;
    }
    .blue{
      color: ■ blue;
    }
  </style>
</head>
```

```

<body>
  <div id="app">
    <h1>{{ titulo}}</h1>
    
    <p :class="classe">Lorem ipsum dolor sit, amet consectetur adipisicing elit.
      Cupiditate placeat illo iste, expedita quos culpa dolor, molestiae quod,
      quibusdam harum quia? Laborum, possimus! Quaerat quis repellendus voluptas sunt, quod repellat!</p>
  </div>

  <script src="https://cdn.jsdelivr.net/npm/vue@2.7.14/dist/vue.js"></script>
  <script>
    var app = new Vue({
      el: '#app',
      data: {
        titulo: 'Aprendendo Vue JS',
        classe: "green"
      }
    })
  </script>
</body>
</html>

```

Agora, vamos detalhar os principais pontos desse código.

1. Diretiva **v-bind** aplicada em atributos HTML dinâmicos

Você está usando a diretiva **v-bind** para atribuir valores dinâmicos a atributos de tags HTML. No caso da imagem (****), você vinculou três atributos:

```



```

Aqui estão os detalhes:

- **v-bind:src="'https://br.vuejs.org/images/logo.png'"**: O **src** da imagem é definido dinamicamente através da diretiva **v-bind**. Neste caso, o valor é uma string estática, mas poderia ser uma variável Vue.
- **v-bind:alt="'Vuejs is awesome!'"**: Da mesma forma, o atributo **alt**, que descreve a imagem para navegadores e ferramentas assistivas, está sendo vinculado a uma string. Isso é útil para descrever a imagem dinamicamente.

- `v-bind:data-img-large="'https://br.vuejs.org/images/logo.png' "`: Este é um exemplo de como você pode usar `v-bind` para atribuir valores a atributos de dados personalizados, como `data-*` attributes. Aqui, você está atribuindo uma URL para `data-img-large`.

2. Interpolação de dados com `{{ titulo }}`

No elemento `<h1>`, você usou a sintaxe de interpolação de dados do Vue.js para exibir dinamicamente o valor da variável `titulo`:

```
<h1>{{ titulo }}</h1>
```

O valor de `titulo` vem do objeto `data` no Vue:

```
data: {  
  titulo: 'Aprendendo Vue JS',  
}
```

Quando o Vue.js carrega, ele substitui `{{ titulo }}` pelo valor `Aprendendo Vue JS`, exibindo-o na página.

3. Ligação dinâmica de classes com `:class`

No parágrafo (`<p>`), você usou a diretiva `v-bind` para dinamicamente adicionar uma classe ao elemento:

```
<p :class="classe">
```

Neste caso, a classe `green` é aplicada ao parágrafo, que corresponde à regra CSS definida no `<style>` do seu documento:

```
<style>  
  .red{  
    color: red;  
  }  
  .green{  
    color: green;  
  }  
  .blue{  
    color: blue;  
  }  
</style>
```

Isso faz com que o texto dentro do parágrafo seja exibido na cor verde. Se você quisesse alterar dinamicamente a classe, bastaria mudar o valor da variável `classe` dentro do `Vue.js`.

4. Instância `Vue.js`

No script, você criou uma instância `Vue` e associou ela ao elemento HTML com `id="app"`

```
<script>
  var app = new Vue({
    el: '#app',
    data: {
      titulo: 'Aprendendo Vue JS',
      classe: "green"
    }
  })
</script>
```

`el: '#app'` vincula a instância `Vue` ao elemento HTML com o ID `app`, ou seja, tudo dentro do `<div id="app">` será controlado pelo `Vue`.

`data` é um objeto onde você define as variáveis reativas, ou seja, qualquer alteração nesses valores será refletida automaticamente na página. Aqui você tem as propriedades `titulo` e `classe`.

Resumo final:

- O valor do atributo `src`, `alt` e `data-img-large` da imagem está sendo atribuído de forma dinâmica usando a diretiva `v-bind`, embora no seu exemplo os valores estejam fixos como strings. Em um caso real, essas strings poderiam ser variáveis do `Vue`.
- O texto do título (`<h1>`) está sendo renderizado dinamicamente usando interpolação de dados com `{{ titulo }}`, e o valor inicial é `'Aprendendo Vue JS'`.
- O parágrafo (`<p>`) recebe a classe CSS dinamicamente através do `:class`, e no exemplo a classe aplicada é `green`, fazendo o texto ser exibido em verde.

Resultado final do projeto:

Aprendendo Vue JS



Lorem ipsum dolor sit, amet consectetur adipisicing elit. Cupiditate placeat illo iste, expedita quos culpa dolor, molestiae quod, quibusdam harum quia? Laborum, possimus! Quaerat quis repellendus voluptas sunt, quod repellat!