



## Modificadores

### Transcrição

Podemos deixar nossa diretiva mais bem organizada. Por exemplo, hoje passamos um objeto com dois parâmetros para nossa diretiva. Mas veja que um parâmetro é usado para controlarmos se realizamos animações ou não. Nesse caso, podemos usar um modificador com esta finalidade.

Por exemplo, nossa diretiva em uso ficará assim:

```
<!-- alurapic/src/components/home/Home.vue -->
```

```
<imagem-responsiva :url="foto.url" :titulo="foto.titulo" v-  
meu-transform.animate="15"/>
```

[COPIAR CÓDIGO](#)

Veja que usamos o nome da diretiva seguido de ponto e o nome do modificador que estamos usando.

Agora, na declaração da nossa diretiva, podemos acessar `binding.modifiers` que é um objeto JavaScript cuja as propriedades são nossos modificadores.

```
import Vue from 'vue';  
  
Vue.directive('meu-transform', {  
  
  bind(el, binding, vnode) {  
  
    let current = 0;
```

```
el.addEventListener('dblclick', function() {  
  
  let incremento = binding.value || 90;  
  
  current+=incremento;  
  this.style.transform = `rotate(${current}deg)`;  
  if (binding.modifiers.animate) this.style.transition  
= "transform 0.5s";  
  
  });  
}  
  
});
```

[COPIAR CÓDIGO](#)

O valor de `binding.modifiers.animate` é sempre `true` quando o modificador estiver presente. Podemos ter mais de um modificador se assim desejarmos. Por exemplo, vamos adicionar o modificador `reverse` para que a imagem gire em sentido anti horário:

```
<!-- alurapic/src/components/home/Home.vue -->
```

```
<imagem-responsiva :url="foto.url" :titulo="foto.titulo" v-  
meu-transform.animate.reverse="15"/>
```

[COPIAR CÓDIGO](#)

Alterando o código da nossa diretiva para levar em consideração o novo modificador:

```
// alurapic/src/directives/Transform.js
```

```
import Vue from 'vue';
```

```
Vue.directive('meu-transform', {
```

```
bind(el, binding, vnode) {  
  
  let current = 0;  
  
  el.addEventListener('dblclick', function() {  
  
    let incremento = binding.value || 90;  
  
    if(binding.modifiers.reverse) {  
      current-=incremento;  
    } else {  
      current+=incremento;  
    }  
  
    this.style.transform = `rotate(${current}deg)`;  
    if (binding.modifiers.animate) this.style.transition  
= "transform 0.5s";  
  
    });  
  }  
  
});
```

[COPIAR CÓDIGO](#)