

Como Funciona o **ElementRef** em Diretivas no Angular

O **ElementRef** no Angular permite acessar e manipular diretamente os elementos do DOM dentro de **diretivas**. Isso é útil quando precisamos alterar estilos, adicionar classes, focar inputs ou até mesmo modificar o conteúdo de um elemento.

1 O que é **ElementRef**?

O **ElementRef** é um serviço do Angular que dá acesso ao elemento DOM no qual a diretiva foi aplicada. Ele permite modificar diretamente o elemento, semelhante ao `document.querySelector()` no JavaScript.

Cuidado!

A manipulação direta do DOM pode expor seu código a vulnerabilidades de segurança, então **prefira usar **Renderer2** sempre que possível** para evitar problemas como ataques XSS.

2 Exemplo Básico: Alterando o Estilo de um Elemento

Vamos criar uma diretiva que altera o **fundo e a cor do texto** do elemento ao qual ela for aplicada.

Código da Diretiva (highlight.directive.ts**)**

```
import { Directive, ElementRef, OnInit } from '@angular/core';
```

```
@Directive({  
  selector: '[appHighlight]' // Essa diretiva será aplicada a qualquer elemento  
  que tenha esse atributo  
})  
export class HighlightDirective implements OnInit {
```

```
constructor(private el: ElementRef) {} // Injetamos o ElementRef no construtor
```

```
ngOnInit() {  
  // Acessamos o elemento diretamente e alteramos seu estilo  
  this.el.nativeElement.style.backgroundColor = 'yellow';  
  this.el.nativeElement.style.color = 'black';  
  this.el.nativeElement.style.padding = '10px';  
}  
}
```

Uso no Template

`<p appHighlight>`Esse parágrafo será destacado em amarelo.`</p>`

 **Quando essa diretiva for aplicada ao `<p>`, o fundo dele ficará amarelo automaticamente.**

3 Exemplo: Alterando o Conteúdo do Elemento

Podemos também modificar o **texto interno** do elemento.

```
import { Directive, ElementRef, OnInit } from '@angular/core';  
  
@Directive({  
  selector: '[appTextChanger]'  
})  
export class TextChangerDirective implements OnInit {  
  
  constructor(private el: ElementRef) {}  
  
  ngOnInit() {  
    this.el.nativeElement.textContent = 'Texto alterado pela Diretiva!';  
  }  
}
```

Uso no Template

<h1 appTextChanger>Esse texto será alterado pela diretiva!</h1>

✓ O texto original será substituído automaticamente.

4 Exemplo: Focar Automaticamente em um Input

Se quisermos que um **campo de entrada (input)** receba foco automaticamente assim que a página carregar, podemos usar **ElementRef** para isso.

```
import { Directive, ElementRef, OnInit } from '@angular/core';
```

```
@Directive({
  selector: '[appAutoFocus]'
})
export class AutoFocusDirective implements OnInit {

  constructor(private el: ElementRef) {}

  ngOnInit() {
    this.el.nativeElement.focus();
  }
}
```

Uso no Template

```
<input appAutoFocus placeholder="Esse input receberá foco automaticamente!">
```

✓ Quando a página carregar, o input será focado automaticamente.

5 Exemplo: Adicionando uma Classe CSS ao Elemento

Podemos usar **ElementRef** para adicionar dinamicamente classes ao elemento.

```
import { Directive, ElementRef, OnInit } from '@angular/core';

@Directive({
  selector: '[appAddClass]'
})
export class AddClassDirective implements OnInit {

  constructor(private el: ElementRef) {}

  ngOnInit() {
    this.el.nativeElement.classList.add('minha-classe-dinamica');
  }
}
```

Uso no Template

`<div appAddClass>`Essa div terá uma classe CSS adicionada dinamicamente!`</div>`

Definição no CSS

```
.minha-classe-dinamica {
  font-size: 20px;
  font-weight: bold;
  border: 2px solid red;
}
```

 **A `div` ganhará automaticamente a classe `minha-classe-dinamica`, aplicando os estilos definidos no CSS.**

Exemplo: Capturando o Evento de Clique

Podemos detectar quando o usuário clica em um elemento e alterar seu estilo.

```
import { Directive, ElementRef, HostListener } from '@angular/core';
```

```
@Directive({
  selector: '[appClickEffect]'
})
export class ClickEffectDirective {

  constructor(private el: ElementRef) {}

  @HostListener('click') onClick() {
    this.el.nativeElement.style.backgroundColor = 'lightblue';
    this.el.nativeElement.style.color = 'white';
  }
}
```

Uso no Template

```
<button appClickEffect>Clique aqui e veja a mágica!</button>
```

✅ Quando o botão for clicado, sua cor de fundo mudará para azul claro.

Conclusão

O **ElementRef** é uma ferramenta poderosa para manipular o DOM dentro de diretivas Angular. Com ele, podemos: ✅ Modificar estilos (cores, fontes, bordas).

- ✅ Alterar textos internos de elementos.
- ✅ Focar automaticamente em inputs.
- ✅ Adicionar classes dinamicamente.
- ✅ Capturar eventos como **click**, **focus**, etc.

Mas lembre-se: **para maior segurança, prefira **Renderer2** quando possível.** 
