

Exemplos de Uso do ViewChild no Angular

O ViewChild é um decorador poderoso no Angular que permite acessar elementos DOM e componentes filhos de forma programática a partir do template do componente.

Ele oferece diversas funcionalidades, como:

- **Manipular propriedades, eventos e métodos de elementos DOM:**
 - Atualizar o valor de um campo de input.
 - Aplicar estilos em um elemento HTML.
 - Disparar eventos programaticamente.
 - Obter o valor de um atributo HTML.
- **Interagir com componentes filhos:**
 - Acessar propriedades e métodos de um componente filho.
 - Chamar funções do componente filho.
 - Passar dados entre o componente pai e o filho.
- **Criar diretivas personalizadas:**
 - Acessar o elemento DOM que está sendo direcionado pela diretiva.
 - Modificar o comportamento do elemento DOM com base em dados da diretiva.
 - Interagir com outros componentes no template.

Casos de Uso Comuns:

1. Acessando e Modificando Elementos DOM:

- **Atualizar valor de um input:**

HTML

```
<input type="text" #meuInput>

@ViewChild('meuInput') meuInputRef: ElementRef;

onClickBotao() {
  this.meuInputRef.nativeElement.value = 'Novo valor';
}
```

- **Aplicar estilos em um elemento:**

HTML

```
<div #meuElemento>Conteúdo do elemento</div>

@ViewChild('meuElemento') meuElementoRef: ElementRef;

onClickBotao() {
  this.meuElementoRef.nativeElement.style.backgroundColor = 'red';
}
```

2. Interagindo com Componentes Filhos:

- **Acessar propriedade de um componente filho:**

HTML

```
<app-meu-componente-filho #meuComponenteFilho></app-meu-componente-filho>

@ViewChild('meuComponenteFilho') meuComponenteFilhoRef: MeuComponenteFilho;

ngOnInit() {
  console.log(this.meuComponenteFilhoRef.nome); // Acessa a propriedade "nome"
  do componente filho
}
```

- **Chamar função de um componente filho:**

HTML

```
<app-meu-componente-filho #meuComponenteFilho></app-meu-componente-filho>

@ViewChild('meuComponenteFilho') meuComponenteFilhoRef: MeuComponenteFilho;

onClickBotao() {
  this.meuComponenteFilhoRef.minhaFuncao(); // Chama a função "minhaFuncao" do
  componente filho
}
```

3. Criando Diretivas Personalizadas:

- **Acessar elemento DOM na diretiva:**

HTML

```
<p appMinhaDiretiva>Texto a ser estilizado</p>
```

```
@Directive({
  selector: '[appMinhaDiretiva]',
})
export class MinhaDiretiva {
  @ViewChild('meuElemento') meuElementoRef: ElementRef;

  ngOnInit() {
    this.meuElementoRef.nativeElement.style.color = 'blue'; // Estiliza o
    texto
  }
}
```

Observações:

- O `ViewChild` só funciona com elementos que estão dentro do template do componente onde ele é declarado.
- Para usar o `ViewChild` com um componente filho, o componente filho precisa ter um seletor de diretiva.
- É importante usar logs de console para verificar se o `ViewChild` está referenciando o elemento ou componente desejado corretamente.

Recursos Adicionais:

- Documentação Angular sobre `ViewChild` [URL inválido removido]
- [Exemplo de uso de `ViewChild`](#)
- [Vídeo sobre `ViewChild` no Angular](#)

Espero que estes exemplos tenham demonstrado a versatilidade do `ViewChild` e suas diversas aplicações no desenvolvimento Angular. Se você tiver mais dúvidas ou precisar de exemplos específicos para o seu caso de uso, não hesite em perguntar!

