

A **interpolação** no Angular é uma técnica usada para exibir dados das propriedades de uma classe no HTML. Ela é representada por `{{ }}` e permite vincular dados da lógica do componente diretamente ao template.

Como funciona:

A interpolação avalia expressões JavaScript no contexto do componente e exibe o resultado como uma string.

Exemplos básicos:

1. Exibir texto de uma propriedade:

```
// app.component.ts
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  template: `<h1>{{ title }}</h1>`, // Exibe o título no HTML
})
export class AppComponent {
  title = 'Bem-vindo ao Angular 19!';
}
```

Saída no navegador:

Bem-vindo ao Angular 19!

2. Usar expressões dentro da interpolação:

Você pode realizar cálculos ou concatenar strings diretamente:

```
// app.component.ts
@Component({
  selector: 'app-root',
  template: `<p>A soma de 2 + 3 é: {{ 2 + 3 }}</p>`,
})
```

```
  })  
  export class AppComponent {}
```

Saída no navegador:

A soma de 2 + 3 é: 5

3. Interpolar propriedades de objetos:

```
@Component({  
  selector: 'app-root',  
  template: `<p>Olá, {{ user.firstName }} {{ user.lastName  
}}!</p>`,  
})  
export class AppComponent {  
  user = { firstName: 'Jessica', lastName: 'Portilio' };  
}
```

Saída no navegador:

Olá, Jessica Portilio!

4. Métodos do componente:

A interpolação também pode chamar métodos simples que retornam valores.

```
@Component({  
  selector: 'app-root',  
  template: `<p>Hoje é: {{ getDate() }}</p>`,  
})  
export class AppComponent {  
  getDate() {  
    return new Date().toLocaleDateString();  
  }  
}
```

Saída no navegador (depende da data atual):

Copiar código

Hoje é: 28/12/2024

5. Trabalhando com arrays:

```
@Component({
  selector: 'app-root',
  template: `<p>0 primeiro item da lista é: {{ items[0]
}}</p>`,
})
export class AppComponent {
  items = ['Angular', 'React', 'Vue'];
}
```

Saída no navegador:

0 primeiro item da lista é: Angular

O que pode ser feito com interpolação:

Interpolar estilos: Use para definir classes ou estilos diretamente:

```
@Component({
  selector: 'app-root',
  template: `<p style="color: {{ favoriteColor }}">Texto
colorido</p>`,
})
export class AppComponent {
  favoriteColor = 'blue';
}
```

1.

Concatenação de strings:

```
<p>{{ 'Olá, ' + name + '!' }}</p>
```

Chamada de funções simples: Como mostrado nos exemplos, chamadas rápidas para retornar dados formatados.

Restrições:

- Não pode ser usada para **atribuições** (exemplo: `{{ nome = 'Jessica' }}` não é permitido).
- Não é recomendada para chamadas de métodos complexos ou com lógica pesada, porque eles são chamados toda vez que o Angular detecta mudanças no componente.