


String Interpolation

```
<h1>Hola my name is {{ name }} and  
I'm {{ age }} years old</h1>  
<h2>{{ 'Hola mundo'.repeat(2) }}  
</h2>  
<p>  
  Las suma de 3 + 3 = {{ 3 + 3 }}  
</p>  
<img src={{myImg}} />
```



22

Ahora da feedback de



Curso de Fundamentos de Angular



→|PROPERTY BINDI...

String interpolation 6/22



RECURSOS

MARCADORES

String interpolation es la manera de enviar datos desde nuestro componente hacia la vista. Utilizando el doble símbolo de llaves `{{ }}`, o también conocidos como brackets, puedes imprimir el valor de

una variable, realizar operaciones matemáticas o hacer el llamado a una función dentro del código HTML.

```
<h1>{{ 'Hola Platzi' }}</h1>
<h2>1 + 1 = {{ 1 + 1 }}</h2>
<h3>{{ myFunction(); }}</h3>
```

División de responsabilidad

Un componente de Angular se divide en tres archivos: uno para el código **TypeScript**, otro para el código **HTML** y uno más para el código **CSS**.

```
import { Component } from '@angular/core';

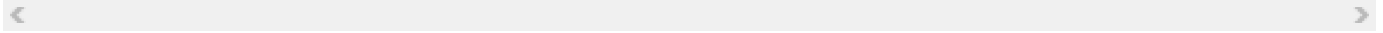
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.scss']
})
export class AppComponent {
  title = 'Hola soy Platzi';
}
```

Angular usa el concepto de **decoradores** para modificar el comportamiento de las clases. La clase `AppComponent` implementa el decorador `@Component()` para indicarle a Angular que esta clase será un componente. Dentro de este decorador, puedes observar el selector del componente (un nombre para el mismo), el template HTML y la hoja de estilos que usará.

Finalmente, dentro de la clase puedes declarar tus propiedades y métodos como en cualquier clase de la programación orientada a objetos. Tenemos una propiedad llamada `title` que es del tipo string. Podemos mostrar el valor de esta variable en el HTML con una interpolación:

```
<p>{{ title }}</p>
```

Es importante que tengas en cuenta la visibilidad de los atributos y métodos de una clase. Si estos llegaran a ser `private`, no podrías usarlo en el HTML. Las variables deben ser públicas para poder ser compartidas al template.



```
...  
export class AppComponent {  
  // Variable privada, no puede utilizarse en un interpolación  
  private title = 'Hola! soy una variable privada';  
}
```



Aporte creado por: Kevin Fiorentino.

Lecturas recomendadas



GitHub - platzi/angular-fundamentals at 4-step
<https://github.com/platzi/angular-fundamentals/tree/4-step>

