



Curso de Angular: Componentes y Servicios



→USO DE INPUTS

¿Qué son los componentes? 2/20



RECURSOS

MARCADORES

Un **componente** es una pieza de software con una **responsabilidad única** y una **estructura y funcionalidad determinada**, además de ser **reutilizable**.

Es una manera de dividir tu aplicación de una forma escalable para no tener todo en un solo archivo. Por ejemplo, un componente para el header, otro para el footer, uno más para el menú, etc.

Componentes en Angular

Puedes crear tu primer componente en Angular utilizando el comando

```
ng generate component test-name o en su forma corta ng g c test-name .
```

Esta acción creará los siguientes archivos:

- `my-test-name.component.html`
- `my-test-name.component.ts`
- `my-test-name.component.css`
- `my-test-name.component.spec.ts`

1. El archivo `.html` que será el template que tu componente utilizará.
2. El archivo `.ts` que contiene el código TypeScript y la lógica.
3. El archivo `.css` que contiene los estilos.
4. Si escogiste trabajar con un preprocesador de CSS, este archivo puede ser `.scss` , `.sass` o `.less` .

Finalmente, el archivo más extraño, `.spec.ts` que contiene el código de las pruebas unitarias que puedes escribir para automatizar el testing en tu componente.

Angular también importará automáticamente el componente creado en el archivo `app.module.ts` para que automáticamente puedas utilizarlo en tu aplicación.

```
...
import { TestNameComponent } from './test-name/test-name.component';

@NgModule({
  declarations: [
    ...
    TestNameComponent
  ],
```

```
imports: [ ... ],
providers: [],
bootstrap: [AppComponent]
})
export class AppModule { }
```

Partes de un componente Angular

El archivo con la extensión `.ts` es el componente principal de cualquier Angular.

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

@Component({
  selector: 'app-test-name',
  templateUrl: './test-name.component.html',
  styleUrls: ['./test-name.component.scss']
})
export class TestNameComponent {
  ...
}
```

Observa lo más importante, el decorador `@Component()`.

Los decoradores alteran el comportamiento de una clase en Angular, para que el compilador de TypeScript interprete el código de la manera correcta y sepa que una clase es:

- un componente,
- un módulo,
- un servicio,
- una directiva, etc.

Este decorador es quién enlaza el componente con el archivo HTML y la hoja de estilos, además le otorga al componente un **selector** o un nombre para utilizarlo en tus templates.

Contribución creada con los aportes de Kevin Fiorentino.

Lecturas recomendadas



Curso de Fundamentos de Angular [Empieza Gratis] - Platzi
<https://platzi.com/cursos/angular/>



GitHub - platzi/angular-componentes at 1-step
<https://github.com/platzi/angular-componentes/tree/1-step>



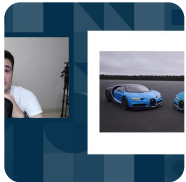
Clases relacionadas



Qué son los componentes



Curso de Frameworks y Librerías de JavaScript



¿Qué son los Web Components?



Curso Básico de Web Components con JavaScript