

# Creación de Componentes en Angular

Los componentes son la base de las aplicaciones Angular. Este tutorial te guiará paso a paso en la creación de componentes efectivos para tu proyecto.





# ¿Qué es un Componente Angular?

## Definición

Un componente es un bloque de construcción de la interfaz de usuario en Angular.

## Estructura

Consta de una clase TypeScript, un template HTML y estilos CSS opcionales.

## Función

Encapsula la lógica y la vista de una parte específica de la aplicación.

# Generación de Componentes con Angular CLI

1

## Abrir Terminal

Navega hasta el directorio raíz de tu proyecto Angular.

2

## Ejecutar Comando

Utiliza "ng generate component nombre-del-componente" para crear un nuevo componente.

3

## Verificar Creación

Comprueba que se han generado los archivos del componente en la estructura del proyecto.







# Estructura de Archivos del Componente

1

## Archivo TypeScript (.ts)

Contiene la lógica del componente y su decorador `@Component`.

2

## Template HTML (.html)

Define la estructura y el contenido visual del componente.

3

## Estilos CSS (.css)

Archivo opcional para estilos específicos del componente.

4

## Archivo de Pruebas (.spec.ts)

Contiene las pruebas unitarias para el componente.

# Anatomía del Archivo .ts del Componente

## Importaciones

Importa las dependencias necesarias de Angular y otros módulos.

## Decorador `@Component`

Define metadatos como el selector, template y estilos del componente.

## Clase del Componente

Implementa la lógica del componente, incluyendo propiedades y métodos.



# Configuración del Template HTML

1

## Estructura Básica

Define la estructura HTML del componente utilizando etiquetas estándar.

2

## Enlace de Datos

Utiliza sintaxis de enlace como `{{ }}` para mostrar datos dinámicos.

3

## Directivas

Implementa lógica en el template con directivas como `*ngIf` y `*ngFor`.





# Estilos del Componente



## Estilos Locales

Define estilos específicos para el componente en el archivo .css.



## Estilos Globales

Aplica estilos globales desde el archivo styles.css del proyecto.



## Encapsulación

Configura la encapsulación de estilos en el decorador @Component.



# Uso y Reutilización de Componentes

Selector

Utiliza el selector definido para incluir el componente en otros templates.

@Input()

Permite pasar datos desde el componente padre al hijo.

@Output()

Emite eventos desde el componente hijo al padre.