

Instalar Angular CLI

Angular CLI (Command Line Interface) es una herramienta que facilita la creación y gestión de aplicaciones Angular.

- Para instalar Angular CLI globalmente, usa el siguiente comando en tu terminal:

```
npm install -g @angular/cli
```

Para comprobar que Angular CLI se ha instalado correctamente, puedes ejecutar:

```
ng version
```

Crear un nuevo proyecto Angular

Una vez que tengas instalado Angular CLI, puedes crear un nuevo proyecto con este comando:

```
ng new nombre-del-proyecto
```

Para ejecutar la aplicación, usa el comando:

```
ng serve
```

Esto levantará un servidor local y podrás ver tu aplicación Angular en el navegador en la dirección: <http://localhost:4200>.

5. Explorar la estructura del proyecto

Una vez creado el proyecto, te recomiendo echar un vistazo a los archivos y carpetas generados:

- **src/app**: Aquí es donde se encuentra el código principal de tu aplicación.
- **app.component.ts**: Es el archivo que define el primer componente de tu aplicación.
- **app.module.ts**: Es el archivo donde se configuran los módulos y componentes que forman parte de tu aplicación.

Hacer un primer cambio

Puedes abrir el archivo `app.component.html` y hacer un pequeño cambio para verificar que todo está funcionando correctamente.

1. Abre `src/app/app.component.html`.
2. Cambia el texto dentro del archivo, por ejemplo:

```
<h1>¡Hola, Angular!</h1>
```

Crear un componente

`ng generate component nombre-del-componente`

Ejemplo:

Si quieres crear un componente llamado `contact-card`, el comando sería:

Esto generará los siguientes archivos en una carpeta llamada `contact-card`:

```
src/app/contact-card/  
├── contact-card.component.ts  
├── contact-card.component.html  
├── contact-card.component.css  
└── contact-card.component.spec.ts
```

contact-card.component.ts: Contiene la lógica del componente (la clase TypeScript).

contact-card.component.html: El archivo de plantilla donde defines el HTML del componente.

contact-card.component.css: El archivo de estilos CSS asociado al componente.

contact-card.component.spec.ts: El archivo para pruebas unitarias del componente (si decides usar pruebas unitarias).

Crear un componente sin archivos de prueba (**spec**):

`ng g component nombre-del-componente --skip-tests`

Crear un componente standalone (autónomo):

A partir de Angular 14, puedes crear un **componente standalone** (sin necesidad de declararlo en un módulo):

`ng g component nombre-del-componente --standalone`

Incluir Bootstrap usando npm

Este método es más robusto porque instala Bootstrap como una dependencia local en tu proyecto Angular.

Paso 1: Instalar Bootstrap y Popper.js usando npm

En la terminal, navega a la raíz de tu proyecto Angular y ejecuta los siguientes comandos para instalar **Bootstrap** y **Popper.js**:

Introducción a Angular

```
npm install bootstrap@5 popper.js --save
```

Paso 2: Incluir Bootstrap en **angular.json**

1. Abre el archivo **angular.json** en la raíz de tu proyecto.
2. Encuentra la sección **"styles"** y **"scripts"**, y agrega las rutas de los archivos CSS y JS de Bootstrap.

```
{
  ...
  "projects": {
    "your-app-name": {
      "architect": {
        "build": {
          "options": {
            "styles": [
              "src/styles.css",
              "node_modules/bootstrap/dist/css/bootstrap.min.css"
            ],
            "scripts": [
              "node_modules/bootstrap/dist/js/bootstrap.bundle.min.js"
            ]
          }
        }
      }
    }
  }
}
```