

# **INTRODUCCIÓN ANGULAR**



- Objetivos aprendidos:
  - Instalar Angular.
  - Crear un proyecto Angular.
  - Ejecutar un proyecto Angular.
  - Sintaxis de Angular.
  - Configuración de módulos y componentes.
  - Utilización de decoradores.
  - Uso de directivas estructurales.
- Explicación:
  - Angular es un framework de Javascript basado en ECMAScript 6, usa Typescript, HTML, CSS y compila a Javascript para poder ser ejecutado. Favorece el uso del patrón Modelo-Vista-Controlador. Es instalado y ejecutado desde la terminal gracias a Angular CLI, que también nos permite lanzar comandos desde la terminal para crear un proyecto nuevo o para crear sus elementos, entre los que encontramos '**Component**', '**Directive**', '**Pipe**', '**Service**', '**Class**', '**Guard**', '**Interface**', '**Enum**', '**Module**'. Angular también nos permite la utilización de testing. Está basado en componentes webs los cuales incluyen la clase CSS, HTML y TS, estos componentes nos permiten una reutilización de código en muchos otros proyectos.
  - Angular está compuesto por diferentes elementos, entre los que se encuentran:
    - **Módulo:** Es un conjunto de códigos dedicados a resolver un objetivo que generalmente exporta una clase. Cada clase posee al menos una clase de módulo que denominamos 'módulo raíz', que, por defecto, posee el nombre de **AppModule**. En este podemos ver una clase con un decorador '@NgModule' y una serie de propiedades como **Declarations**, **Exports**, **Imports**, **Providers** y **Bootstrap**.
    - **Componente:** Este elemento controla una zona de espacio de la pantalla que representa la 'vista'. Define las propiedades y métodos que usa el propio template y contiene la lógica y la funcionalidad que utiliza la vista. Pueden tener atributos de entrada y de salida.
    - **Template:** Define la vista de un componente mediante un fragmento HTML.
    - **Metadatos:** Permiten configurar el comportamiento de una clase a la que se le ha asociado un decorador.
    - **Data binding:** Permite intercambiar datos entre el template y la clase del componente. Hay 4 tipos de este elemento, **Interpolación**, **Property binding**, **Event binding** y **Two-way binding**.
    - **Directiva:** Permite añadir comportamiento dinámico a HTML. Para ello podemos usar '**\*ngFor**' para introducir un bucle, '**\*ngIf**' para añadir un condicional, o incluso podemos modificar el aspecto o comportamiento de un elemento **DOM** con '**NgSwitch**', '**NgStyle**' y '**NgClass**'.
    - **Servicio:** Son clases que permiten realizar acciones concretas que ponen a disposición de cualquier componente.

- **Dependency Injection:** Suministra funcionalidades, servicios y demás a un componente sin que sea el propio componente el encargado de crearlos. Utiliza el decorador '@Injectable()'.
- En el módulo '**app.module.ts**' podemos apreciar diferentes partes:
  - Los **import**, que están compuestos de **BrowserModule, NgModule y AppComponent**.
  - El decorador **@NgModule** con su metadata que viene a ser **Declarations, Imports, Providers y Bootstrap**.
  - A su clase, que permite cargar la aplicación.
- Este archivo agrupa todas las partes descritas anteriormente que constituyen el módulo en sí.
- Los componentes se dividen en 3 partes que constituyen el Modelo-Vista-Controlador. Generalmente un componente lleva asociado:
  - **app.component.html** (vista).
  - **app.component.css**
  - **app.component.ts** (controlador)
  - **app.component.spec.ts** (usado para test unitarios)
- Debemos de tener en cuenta que los componentes que creamos en Angular son como pequeñas piezas de un puzzle que, una vez que queramos verlo en la página HTML correcta, deberemos ir exportando e importando los elementos hasta llegar al HTML final. Un ejemplo de esto sería como una fábrica de coches, en cada sector se crea una determinada pieza del coche para finalmente unir las todas en un único destino y montarlo.
- Con toda esta información más la que podamos encontrar en la misma página oficial, tutoriales, libros, etc., podremos empezar a trabajar en nuevos proyectos.
- Proyectos creados:
  - He creado un proyecto con guía de la página oficial de Angular. El enlace a github para este proyecto es el siguiente:
    - <https://github.com/Joseatr/2DAW-DWEC/tree/master/Trabajos/Angular/tourOfHeroes>
- Fuentes bibliográficas consultadas:
  - Página oficial de Angular (<https://angular.io/>)
  - El Gran Libro de Angular.