# Servicios en Angular

Las aplicaciones en Angular se desarrollan en base un árbol de componentes. A medida que un proyecto vaya creciendo lo ideal es organizar bien el código manteniendo piezas de responsabilidad reducida. En ocasiones también nos puede suceder que dos o más componentes tengan que acceder a los mismos datos los que nos puede llevar a repetir código. Para solucionar esto existen los servicios.

Básicamente **un servicio es un proveedor de datos, que mantiene la lógica de acceso a ellos.** Los servicios van a ser consumidos por los componentes y delegaran en ellos la responsabilidad de acceder a la información y realizar operaciones con ellos.

## Crear un servicio

Para crear un servicio mediante Angular-CLI hacemos lo siguiente:



En este caso nos va a crear un servicio llamado Ejemplo1Servcie dentro de la carpeta services (el sufijo *service* lo agrega Angular-CLI).

**Nota:** como los servicios suelen ser utilizados desde varios componentes, a veces se estila guardarlos dentro de una carpeta llamada *shared, común* o algo parecido.

## Agregar la declaración de un servicio a un módulo

Para poder utilizar el servicio es necesario agregarlo a un módulo. Una vez agregado lo vamos a poder utilizar desde cualquiera de los componentes que pertenecen a ese modulo.

**Nota:** al contrario de lo que pasa cuando creamos con Angular-CLI un componente, cuando creamos un servicio este no se añade automáticamente en el módulo donde lo estamos creando por lo cual lo debemos declarar manualmente.

Para hacerlo dentro de los parámetros del decorador del módulo @NgModule, en el array *provider*s (previamente debemos hacer el import del servicio) Quedaría de la siguiente manera:

Importamos el servicio



Importamos el servicio



**Nota:** de esta manera agregamos un servicio para que esté disponible para todos los componentes. También es posible agregarlo solo para un componente en particular.

## Código básico de un service en Angular



En la primera línea se importa el decorador Injectable para poder utilizarlo más abajo

El decorador @Injectable indica que la clase que se decora, en este caso Ejemplo1Service, puede necesitar dependencias que pueden ser provistas por inyección de dependencias.

La clase Ejemplo1Service no es distinta a cualquier otra clase, podemos declarar atributos y métodos.

## Como inyectar dependencias de servicios

En Angular (y en muchos otros frameworks**) la inyección de dependencias se realiza por medio del constructor**. En el constructor de un componente podemos declarar cualquiera de los servicios que vayamos a usar:



De esta manera le estamos indicando a TypeScript y a Angular que vamos a usar un objeto *miEjemploService* de es del tipo *Ejemplo1Service.* A partir de ese momento dentro del componente va existir ese objeto proporcionando todos los datos y funcionalidad definida en el servicio.

**Nota:** nótese que en la declaración de la inyección de dependencias en el constructor lleva la declaración de visibilidad (public o private)

## Explicando con detalle la declaración de propiedades implícita en el constructor

Aquí tenemos que detenernos para explicar algo de la magia que ofrece TypeScript porque el hecho de que declare una propiedad en un objeto solo porque se reciba un parámetro en el constructor no es algo usual en JavaScript.

Cuando TypeScript detecta el modificador de visibilidad *public* o *private* en el parámetro definido en el constructor, inmediatamente declara una propiedad en la clase y le asigna el valor recibido en el constructor.  
Por lo cual el siguiente código:



Es equivalente a:



En resumen, cuando TypeScript encuentra un parámetro de visibilidad en el constructor entiende que lo que quieres hacer en realidad es crear una propiedad en el objeto recién construido con el valor recibido como parámetro.

## Utilizando el servicio

### Dentro de la clase del componente

Dentro de la clase de nuestro componente tenemos el servicio disponible a partir de la propiedad usada en su declaración.



### Dentro de la template