# Pipes

Un **pipe** (o "tubería") en Angular es una **función especial que transforma datos** directamente en el template (HTML), sin tener que modificarlos en el componente.

## 📦 Pensalo así:

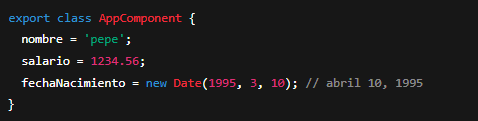
Es como un **filtro o formateador de datos**. Le das un valor crudo (ej: número, fecha, string) y te lo devuelve transformado (ej: con formato de moneda, mayúsculas, fecha legible, etc.).

## 🛠️ ¿Para qué sirve?

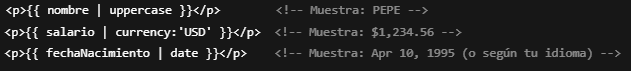
* Formatear **números**
* Mostrar **fechas** bien escritas
* Convertir **textos** a mayúsculas o minúsculas
* Formatear como **moneda** o **porcentaje**
* Traducir, filtrar, ordenar, etc. (¡también se pueden crear pipes personalizados!)

## Ejemplo

Supongamos este componente:



En el HTML:



🧩 Sintaxis general:



Ejemplo con parámetros:



## 🧠 Ventajas de usar pipes

* Son **fáciles de usar**
* No cambian el valor original
* Mejoran la **presentación** sin lógica extra
* Podés **encadenarlos**:



# Tipos de pipe

## Lowercase Pipe



Transforma todo el string a minúscula

## Uppercase Pipe

Transforma todo el string a mayúscula



## Titlecase Pipe



Convierte el texto a mayúsculas y minúsculas. Pon en mayúscula la primera letra de cada palabra y el resto a minúsculas. Las palabras se delimitan con cualquier espacio en blanco, como un espacio, una tabulación o un salto de línea.

## Date Pipe

Edita la fecha para mostrarla en el formato especificado



Ejemplo con parámetro de formato



## DecimalPipe o NumberPipe

Formatea un valor según las opciones de dígitos y las reglas de configuración regional.

(Según la región por “,” o “.”)

Ej

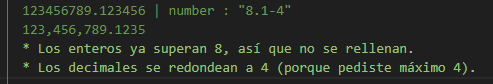


### 1er parametro completa enteros con 0 hasta llegar a esa cantidad



### 2do parametro y 3er parámetro

cuantos decimales mínimos queres mostrar y máximo



## Currency Pipe

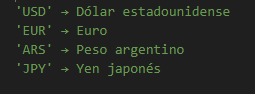
Muestra en formato moneda



### 1er parámetro: moneda (opcional)

Es el código ISO 4217 de la moneda.

 primeras 2 letras del pais y una letra de la moneda (opcional)



### 2do parámetro: display (opcional)

Controla como se muestra la moneda

Valores : ‘symbol’ and ‘code’

Symbol : símbolo de la moneda solo : $

Code : código de la moneda : ARS, EUR

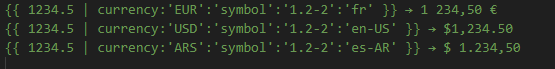
### 3er parámetro digitsinfo (opcional)

Funciona igual que el pipe number con los decimales

### 4to parámetro

Controla la forma regional de mostrar el número y la moneda.

Ej:



## Percent pipe



### 1er parametro funciona igual que el pipe number

## I18nSelectPipe

Selector genérico que muestra la cadena que coincide con el valor actual.

Si ninguna de las claves de mappingcoincide con value, se devuelve el contenido de la otherclave cuando está presente, de lo contrario, se devuelve una cadena vacía.

@Component({

selector: 'i18n-select-pipe',

template: `<div>{{ gender | i18nSelect: inviteMap }}</div>`,

standalone: false,})

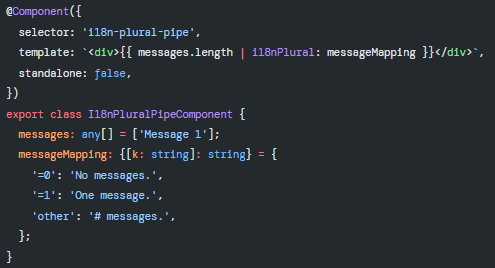
export class I18nSelectPipeComponent {

gender: string = 'male';

inviteMap: any = {'male': 'Invite him.', 'female': 'Invite her.', 'other': 'Invite them.'};}

## I18nPluralPipe

Asigna un valor a una cadena que pluraliza el valor según las reglas de configuración regional.



## SlicePipe

Crea un conjunto nuevo Array o String que contiene un subconjunto (sección) de los elementos.



El primer parámetro incluye el elemento, el 2do no

También se puede empezar con números negativos

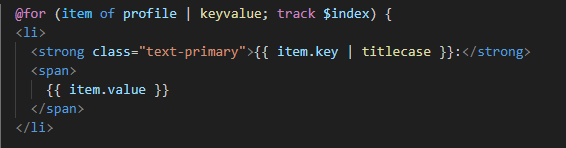
 el -3 empezará desde atrás del array

## JsonPipe

Convierte un valor a su representación en formato JSON. Útil para la depuración.



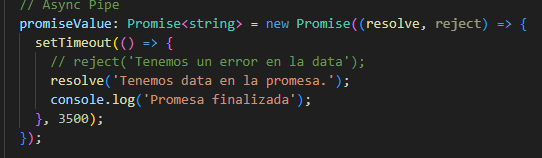
## KeyValuePipe

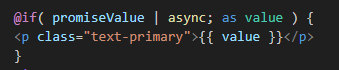
Transforma un objeto en un array de pares clave-valor.  


## AsyncPipe

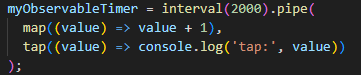
Desenvuelve un valor de una primitiva asíncrona.

La tubería asíncrona se suscribe a un Observable o Promesa y devuelve el último valor emitido. Al emitirse un nuevo valor, la tubería asíncrona marca el componente para que se comprueben los cambios. Cuando el componente se destruye, la tubería asíncrona se cancela automáticamente para evitar posibles fugas de memoria. Cuando cambia la referencia de la expresión, la tubería asíncrona se cancela automáticamente de la suscripción al Observable o Promesa anterior y se suscribe al nuevo.





Solo valores emitidos (no atrapa errores/reject)



# Pipes personalizados

Crear pipe

Ng g pipes|p

