

# DESARROLLO DE APLICACIONES CON ANGULAR

Angular

1



# ANGULAR

## ■ Pipes

- Permiten realizar transformaciones de elementos de forma automática.
- Los pipes se aplican utilizando el símbolo | entre los datos y el pipe de transformación.
- Es un elemento de transformación de la salida, pero no modifican el contenido transformado.

<https://angular.io/guide/pipes>  
<https://angular.io/api/common#pipes>



# ANGULAR

- Pipes-JsonPipe
  - Permite mostrar objetos en formato JSON.
  - <https://angular.io/api/common/JsonPipe>

```
{{ empleado | json }}
```



# ANGULAR

## ■ Pipes-DecimalPipe

- Permite expresar el formato de los números decimales.
- <https://angular.io/api/common/DecimalPipe>

```
<p>{{ salario | number : '6.2-3' }}</p>
```



# ANGULAR

- Pipes-Currency
  - Permite representar monedas.
  - <https://angular.io/api/common/CurrencyPipe>

```
<p>{{ salario | currency:'EUR':'symbol':'6.2-3'}}</p>
```



# ANGULAR

## ■ Pipes-Date

- Permite representar fechas.
- <https://angular.io/api/common/DatePipe>

```
<p>{{ hoy | date : 'dd-MM-YYYY' }}</p>
```



# ANGULAR

## ■ Pipes-Percent

- Permite representar porcentajes.
- <https://angular.io/api/common/PercentPipe>

```
<p>{{ iva | percent : '2.2-3' }}</p>
```



# ANGULAR

- Pipes-UpperCase-LowerCase-TitleCase
  - Permite mostrar textos en mayúsculas, minúsculas y en modo 'título'.
  - <https://angular.io/api/common/UpperCasePipe>
  - <https://angular.io/api/common/LowerCasePipe>
  - <https://angular.io/api/common/TitleCasePipe>

`<p>{{ texto | titlecase }}</p>`





# ANGULAR

## ■ Pipes-Slice

- Permite mostrar secciones de colecciones.
- <https://angular.io/api/common/SlicePipe>

```
<ul>
  <li *ngFor="let nombre of nombres | slice:0:2">
    {{nombre}}
  </li>
</ul>
```



# ANGULAR

## ■ Pipes-Async

- Permite invocar directamente a métodos asíncronos.
- <https://angular.io/api/common/AsyncPipe>
- HTML

```
{{ promesa | async }}
```

## ■ TS

```
promesa: Promise<any> | null = null;  
this.promesa = new Promise((resolve, reject) => {  
  setTimeout(() => {  
    console.log("Resolviendo...");  
    resolve!("¡OFERTA FLASH!");  
  }, 5000);  
});
```



# ANGULAR

## ■ Pipes-Key Value

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

- <https://angular.io/api/common/KeyValuePipe>

- HTML

```
{{ input | keyvalue }}
```

- TS

```
<select>
```

```
  <option *ngFor="let item of nombresDiasSemana | keyvalue"  
    value={{item.key}}>{{item.value}}</option>
```

```
</select>
```



# ANGULAR

## ■ Pipes-I18nPlural

- Permite asignar la representación singular o plural de un mensaje.

- <https://angular.io/api/common/I18nPluralPipe>

- HTML

```
<div>{{ nombresDiasSemana.length | i18nPlural: messageMapping }}
```

- TS

```
nombresDiasSemana = ["lunes", "martes", "miércoles", "jueves", "viernes", "sábado", "domingo"];
```

```
messageMapping:
```

```
{[k: string]: string} = {'=0': 'Ningún día.', '=1': 'Un día.', 'other': '# días.'};
```



# ANGULAR

## ■ Pipes-IntlSelectPipe

- Permite seleccionar un texto en función del valor de una expresión.

- <https://angular.io/api/common/IntlSelectPipe>

- HTML

```
<div>{{genero | intlSelect: mapaTramamiento}}</div>
```

- TS

```
genero: string = 'sin especificar';
```

```
mapaTramamiento: any = {'hombre': 'Estimado Sr..', 'mujer': 'Estimada Sra.', 'other':  
'Estimad@ client@.'};
```



# ANGULAR

## ■ Pipes

- PIPES PERSONALIZADOS
- `ng generate pipe pipes/nombre-pipe`
- `ng g p pipes/nombre-pipe`



# ANGULAR

## ■ Pipes

- El método tiene como nombre transform y recibe un valor y una sucesión opcional de parámetros
  - transform(value:any)
  - transform(value:string)
  - transform(value:string, args?:any)
- Tiene un retorno, que es el resultado de la transformación de la entrada.
- La invocación se realiza en el código HTML:
  - {{variable | nombre-pipe:parametro1:parametro2:parametro3...}}
- La propiedad del pipe pure determina si el valor que se le pasa es o no modificado.
- @Pipe(
  - name:'nombre',
  - pure:true/false)