



Florida
Universitària

Control Flow: Bucles y condicionales

Curso 2025/2026

Paco Segura

¿Qué es Control Flow?

- Novedad desde Angular 17.
 - Permite utilizar las estructuras de control de flujo **if**, **switch** y **for** en código HTML, extendiendo así sus funcionalidades.
 - Fáciles de usar, esenciales para trabajar en Angular.
 - Vienen para substituir a las directivas *ngFor, *ngSwitch y *ngIf.
-

Condicionales: @if

- **@if:** equivalente a un **if** de JavaScript. Permite renderizar código HTML si se cumple una condición.
- **Sintaxis:**

```
@if (condition) {  
    <div>Title</div>  
}
```

Condicionales: @if

- **Ejemplo:**

Archivo *.html

```
@if (test) {  
  <p>Hello World!</p>  
}
```

Archivo *.ts

```
import { Component } from '@angular/core';  
  
@Component({  
  selector: 'app-ejemplos',  
  imports: [],  
  templateUrl: './ejemplos.html',  
  styleUrls: ['./ejemplos.css'],  
})  
export class Ejemplos {  
  public test: boolean = true;  
}
```

Condicionales: @if

- Podemos añadir un bloque @else:

```
@if (test) {  
  <p>Hello World!</p>  
} @else {  
  <p>Goodbye World!</p>  
}
```

Archivo *.html

```
import { Component } from '@angular/core';  
  
@Component({  
  selector: 'app-ejemplos',  
  imports: [],  
  templateUrl: './ejemplos.html',  
  styleUrls: ['./ejemplos.css'],  
})  
export class Ejemplos {  
  public test: boolean = false;  
}
```

Archivo *.ts

Condicionales: @if

- Podemos añadir lógica con **eventos**:

```
<p (click)="onClick()">Click!</p>
```

Archivo *.html

```
@if (test) {  
  <p>Hello World!</p>  
} @else {  
  <p>Goodbye World!</p>  
}
```

```
export class Ejemplos {  
  public test: boolean = false;  
  
  public onClick(): void {  
    this.test = !this.test;  
  }  
}
```

Archivo *.ts



Condicionales: @switch

- **@switch:** equivalente a un **switch** de JavaScript. Permite renderizar código HTML si se cumple una condición.
- **Sintaxis:**

```
@switch (case) {  
  @case ('one') {  
    <p>Caso uno</p>  
  }  
  @case ('two') {  
    <p>Caso dos</p>  
  }  
  @default {  
    <p>Caso por defecto</p>  
  }  
}
```

Condicionales: @for

- **@for:** equivalente a un **for** de JavaScript. Trabaja como un **forEach** que permite iterar sobre las posiciones de un array, renderizando código HTML con la información que tenemos guardada en cada posición del array.
- **Sintaxis:**

```
@for (item of items; track item) {  
    {{ item }}  
}
```

Condicionales: @for

- **track:** parámetro obligatorio. Al renderizar con **@for**, los elementos del array pueden cambiar según la lógica que tengamos implementada. Angular utiliza track para tener internamente localizados los elementos tras cualquier reordenamiento.
- **Ejemplo:**

```
@for(name of names; track name) {  
  <p>{{name}}</p>  
}
```

Archivo *.html



Condicionales: @for

Archivo *.ts

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-ejemplos',
  imports: [],
  templateUrl: './ejemplos.html',
  styleUrls: ['./ejemplos.css'],
})
export class Ejemplos {
  public names: string[] = ['John', 'Mary', 'Robert', 'Rose'];
}
```

Condicionales: @for

- Muy útil para trabajar con arrays de objetos:

```
@for (item of items; track item.name) {  
  <p> {{ item.name }} </p>  
  <p> {{ item.surname }} </p>  
}
```

Archivo *.html

```
export class Ejemplos {  
  public items: { name: string; surname: string }[] = [  
    { name: 'John', surname: 'Doe' },  
  ];  
}
```

Archivo *.ts

Condicionales: @for

- Tiene varias variables implícitas que podemos usar:
 - **\$index**: número del índice del elemento renderizado.
 - **\$first**: true si es el primer elemento renderizado.
 - **\$last**: true si es el último elemento renderizado.
 - **\$even**: true si el elemento renderizado es par.
 - **\$odd**: true si el elemento renderizado es impar.
-

Condicionales: @for

- **Ejemplo:**

```
@for (student of students; track student; let i = $index; let first = $first; let last = $last;  
let even = $even; let odd = $odd) {  
  <p>{{ student }}</p>  
  <p>{{i}} {{first}} {{last}} {{even}} {{odd}}</p>  
}
```

Archivo *.html

Archivo *.ts

```
export class Ejemplos {  
  public students: string[] = ['John', 'Mary', 'Rose', 'Charles', 'Lucy', 'Alexander'];  
}
```



Condicionales: @for

- Podemos añadir un bloque **@empty** para cuando el array está vacío:

```
@for (student of students; track student) {  
  <p>{{ student }}</p>  
} @empty {  
  <p>Array vacío</p>  
}
```

```
export class Ejemplos { public  
  students: string[] = [];  
}
```
