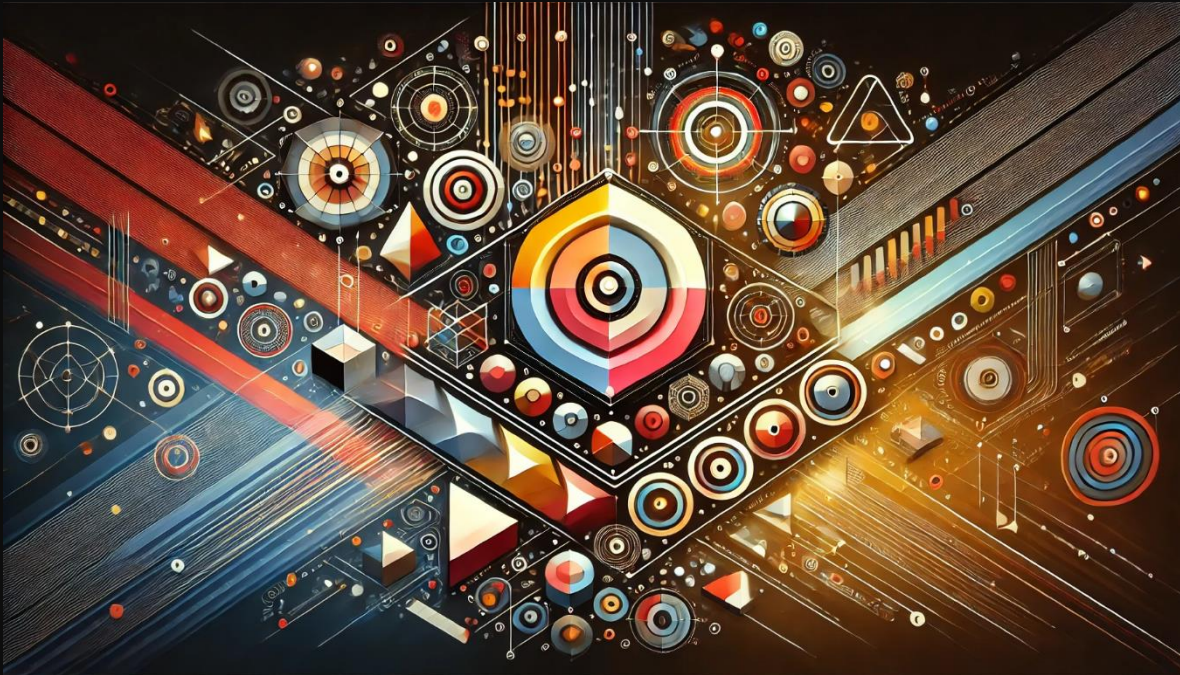


# Directiva @for en Angular



## Controlando el Flujo con @for en Angular

### 1. ¿Qué es @for en Angular?

La directiva @for nos permite iterar sobre colecciones directamente en la plantilla HTML. Esta sintaxis es limpia y expresiva, integrándose muy bien con los bloques de control de flujo como @if y @else. La directiva @for permite generar dinámicamente contenido en la plantilla según los elementos de una colección.

### 2. Sintaxis Básica de @for

La sintaxis básica para usar @for en Angular es la siguiente:

```
@for (elemento of coleccion) {  
  <!-- Código HTML que se repite para cada elemento de la colección -->  
}
```

- **elemento of coleccion:** Declara una variable `elemento` que representa el elemento actual de la iteración y `coleccion` es la colección sobre la que estás iterando.

### 3. Ejemplo Práctico con @for en Angular

Vamos a crear un ejemplo donde se itera sobre una lista de tareas y se muestran en la interfaz.



Primero creamos el componente:

ng g c componente-for --skip-tests

**Componente (tareas.component.ts)**

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

@Component({
  selector: 'app-componente-for',
  standalone: true,
  imports: [],
  templateUrl: './componente-for.component.html',
  styleUrls: ['./componente-for.component.css'],
})
export class ComponenteForComponent {
  tareas: string[] = [
    'Aprender Angular',
    'Desarrollar una App',
    'Aprender TypeScript',
  ];

  agregarTarea(nuevaTarea: string): void {
    if (nuevaTarea) {
      this.tareas.push(nuevaTarea);
    }
  }
}
```

```
}
}
```

### Plantilla (tareas.component.html)

```
<div class="container mt-5">
  <h2 class="text-center">Lista de Tareas</h2>

  <ul class="list-group w-50 mx-auto">
    @for (tarea of tareas; track tarea) {
      <li class="list-group-item text-center">{{ tarea }}</li>
    }
  </ul>

  <div class="mt-3 d-flex justify-content-center">
    <input type="text" #tareaInput class="form-control mb-2 w-50"
      placeholder="Nueva tarea">
  </div>

  <div class="d-flex justify-content-center">
    <button class="btn btn-primary w-50"
      (click)="agregarTarea(tareaInput.value); tareaInput.value=''>
      Agregar Tarea
    </button>
  </div>
</div>
```

## 4. Explicación del Ejemplo

1. **tareas: string[]:**
  - Una propiedad que almacena una lista de tareas como un arreglo de cadenas (string[]).
2. **Directiva @for:**
  - @for (let tarea of tareas):** Itera sobre cada tarea en el arreglo tareas. Por cada tarea, genera un elemento <li> en la lista con el texto de la tarea.
3. **Agregar Tarea:**
  - agregarTarea(nuevaTarea: string):** Este método permite al usuario agregar una nueva tarea a la lista. Se invoca cuando el usuario hace clic en el botón "Agregar Tarea".
4. **Interfaz:**
  - Se usa Bootstrap para estilizar la lista de tareas (list-group) y el formulario de entrada (form-control).

## 5. Resultado Esperado

Al cargar la aplicación, verás una lista de tareas predefinidas. Puedes agregar nuevas tareas ingresando texto en el campo de entrada y haciendo clic en "Agregar Tarea". Las nuevas tareas se agregarán a la lista y se mostrarán inmediatamente gracias a la directiva `@for`.



## 6. Conclusión

La directiva `@for` en Angular simplifica la iteración sobre colecciones en las plantillas, permitiendo generar dinámicamente contenido HTML basado en los elementos de un arreglo. Esta sintaxis es clara y permite una integración fluida con otras directivas como `@if`.

Saludos!

Ing. Ubaldo Acosta

Fundador de [GlobalMentoring.com.mx](http://www.globalmentoring.com.mx)