

Event Binding en Angular



Event Binding en Angular

Event Binding es una técnica que permite enlazar eventos del DOM (Document Object Model) con métodos del componente de Angular. Esto permite que Angular ejecute código en respuesta a la interacción del usuario, como hacer clic en un botón, mover el mouse, escribir en un campo de texto, entre otros.



En Angular, los eventos se manejan mediante **Event Binding**, donde el nombre del evento se coloca dentro de paréntesis y se enlaza a un método o expresión en el componente.

Sintaxis de Event Binding

La sintaxis básica para Event Binding es:

```
<elemento (evento)="metodoDelComponente()"></elemento>
```

- **elemento**: El elemento del DOM (por ejemplo, `button`, `input`, etc.) que disparará el evento.
- **evento**: El nombre del evento del DOM que se desea manejar (por ejemplo, `click`, `keyup`, etc.).
- **metodoDelComponente()**: El método del componente que se ejecutará cuando ocurra el evento.

Ejemplo Básico de Event Binding

Vamos a crear un ejemplo donde se muestra un botón. Cuando el usuario hace clic en el botón, se mostrará un mensaje en la pantalla. Usaremos Bootstrap para darle un estilo más agradable.

En primer lugar, creamos un nuevo componente llamado `MostrarMensajeComponent`:

```
ng generate component mostrar-mensaje --skip-tests
```

Código del Componente (`mostrar-mensaje.component.ts`):

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

@Component({
  selector: 'app-mostrar-mensaje',
  standalone: true,
  imports: [],
  templateUrl: './mostrar-mensaje.component.html',
  styleUrls: ['./mostrar-mensaje.component.css']
})
export class MostrarMensajeComponent {
  mensaje: string = '';

  mostrarMensaje(): void {
    this.mensaje = '¡Hola, has hecho clic en el botón!';
  }

  resetearMensaje(): void {
    // Reiniciamos el valor de la variable
    this.mensaje = '';
  }
}
```

Código del Componente (`mostrar-mensaje.component.html`):

```
<div class="container text-center">
  <div class="d-flex justify-content-center mt-3">
    <button class="btn btn-primary me-2" (click)="mostrarMensaje()">
      Mostrar Mensaje
    </button>
    <button class="btn btn-danger" (click)="resetearMensaje()">
      Resetear Mensaje
    </button>
  </div>
  <div class="alert alert-success mt-4 mx-auto w-50">
    {{ mensaje }}
  </div>
</div>
```

Explicación del Ejemplo:

1. `(click)="mostrarMensaje()"`: Este es el Event Binding. Aquí estamos enlazando el evento `click` del botón al método `mostrarMensaje()` del componente. Cuando el usuario hace clic en el botón, se ejecuta el método `mostrarMensaje()`, que actualiza la propiedad `mensaje`.
2. `mensaje: string = ''`: Inicialmente, `mensaje` está vacío, por lo que no se muestra nada en la pantalla.
3. El botón de `ResetearMensaje` utiliza evento binding para llamar al método `resetearMensaje` al hacer click sobre el botón y coloca nuevamente la variable de `mensaje` a cadena vacía reiniciando el contenido de esta variable.
4. **Bootstrap**: Hemos utilizado clases de Bootstrap para estilizar el contenido. Este es un resumen de las clases Bootstrap utilizadas en este ejemplo:

Aquí tienes un resumen de las clases de **Bootstrap** utilizadas en este código:

Clases de Bootstrap:

Aquí tienes el resumen de las clases de **Bootstrap** con las explicaciones más compactas, manteniendo el detalle de lo que significa cada clase:

- **container**: Crea un contenedor central con márgenes automáticos. Se utiliza para estructurar el contenido dentro de un área delimitada y centrada.
- **text-center**: Alinea el texto al centro del contenedor. Asegura que el contenido, como el texto y los botones, esté centrado horizontalmente.
- **d-flex**: Activa el sistema de diseño **flexbox**. Permite distribuir y alinear los elementos hijos (botones, en este caso) de manera flexible.
- **justify-content-center**: Alinea los elementos flexibles en el centro horizontal del contenedor. Mantiene los botones centrados dentro del espacio disponible.
- **mt-3**: **Margin top (margen superior)** de 3 unidades. Separa los botones del contenido superior.

- **btn btn-primary**: Estilo de botón. **btn** es la clase base para botones, mientras que **btn-primary** le da el color azul, usado para acciones principales.
- **me-2**: **Margin end (margen derecho)** de 2 unidades. Crea espacio a la derecha del botón para separar visualmente ambos botones.
- **btn btn-danger**: Botón con estilo de advertencia o acción destructiva, lo que le da el color rojo. **btn-danger** se usa típicamente para eliminar o restablecer acciones.
- **alert alert-success**: Estilo de alerta con un fondo verde, que indica éxito. **alert** es la clase base para alertas, y **alert-success** añade el color de éxito.
- **mt-4**: **Margin top (margen superior)** de 4 unidades. Proporciona espacio adicional entre la alerta y los botones superiores.
- **mx-auto**: **Margin horizontal automático**. Centra el contenido horizontalmente dentro del contenedor.
- **w-50**: Establece el ancho del elemento en **50%** del contenedor padre, ajustando la alerta para que ocupe solo la mitad del ancho disponible.

Estructura del código:

- **Botones**: Dos botones centrados con un margen entre ellos, usando `d-flex` para alinear y `justify-content-center` para centrar.
- **Mensaje**: Un mensaje dentro de una alerta verde centrada y con un ancho del 50%, usando márgenes automáticos y flexbox para un diseño limpio.

Este código aprovecha las utilidades de **Bootstrap** para crear una interfaz limpia, centrada y con buen espaciado entre los elementos.

Código del Componente (`app.component.ts`):

```
import { Component } from '@angular/core';
import { RouterOutlet } from '@angular/router';
import { NuevoComponenteComponent } from './nuevo-componente/nuevo-componente.component';
import { ComponenteInlineComponent } from './componente-inline/componente-inline.component';
import { PadreComponent } from './padre/padre.component';
import { MostrarMensajeComponent } from './mostrar-mensaje/mostrar-mensaje.component';

@Component({
  selector: 'app-root',
  standalone: true,
  imports: [
    RouterOutlet,
    NuevoComponenteComponent,
    ComponenteInlineComponent,
    PadreComponent,
    MostrarMensajeComponent
  ],
  templateUrl: './app.component.html',
})
```

```
    styleUrls: ['./app.component.css'],
  })
  export class AppComponent {
    titulo = 'Event Binding en Angular';
  }
```

Código de la Plantilla (app.component.html):

```
<section>
  <div class="container text-center">
    <h1 class="text-center text-warning my-4">{{ titulo }}</h1>
    <app-mostrar-mensaje/>
  </div>
</section>
```

Aquí tienes el resumen de las clases de **Bootstrap** usadas en este código con una explicación de cada clase:

- **container**: Crea un contenedor centrado con márgenes automáticos a los lados, utilizado para organizar y delimitar el contenido dentro de una estructura fluida y alineada.
- **text-center**: Alinea todo el contenido dentro del contenedor al **centro** horizontalmente.
- **text-warning**: Aplica un color de texto **amarillo** (de advertencia) al título, útil para destacar información importante o de alerta.
- **my-4**: **Margin vertical (margen superior e inferior)** de 4 unidades. Añade espacio tanto arriba como debajo del título para separar visualmente los elementos.

Resultado Esperado:

Cuando ejecutes la aplicación (`ng serve`), verás dos botones al centro de la pantalla con un diseño agradable gracias a Bootstrap. Al hacer clic en el botón, aparecerá un mensaje de alerta verde con el texto "¡Hola, has hecho clic en el botón!". Y otro botón para Resetear el mensaje y que el texto se elimine de la pantalla.



Conclusión

Event Binding es fundamental en Angular para responder a la interacción del usuario. Este ejemplo simple demuestra cómo se puede enlazar un evento del DOM a un método del componente para actualizar la interfaz de usuario de manera dinámica. El uso de Bootstrap mejora la apariencia visual, haciendo que la aplicación sea más atractiva y fácil de usar.

Para más información de las clases de Bootstrap puedes consultar la documentación oficial:

<https://getbootstrap.com/>

Saludos!

Ing. Ubaldo Acosta

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