

Angular proporciona dos enfoques diferentes para manejar formularios:

1. Formularios basados en plantillas
2. Formularios reactivos

Ambos capturan los eventos de entrada del usuario desde la vista, crean un modelo de formulario y un modelo de datos para actualizar y proporcionan una forma de rastrear los cambios.

Los formularios basados en plantillas son útiles para agregar un formulario simple a una aplicación. Son más fáciles de agregar que los formularios reactivos, pero no escalan tan bien como los formularios reactivos. Si tiene requisitos de formulario y lógica muy básica podemos utilizar sin problemas formularios basados en plantillas.

Veremos en este concepto como implementar en Angular formularios basados en plantillas.

Formulario reactivo (directiva ngModel)

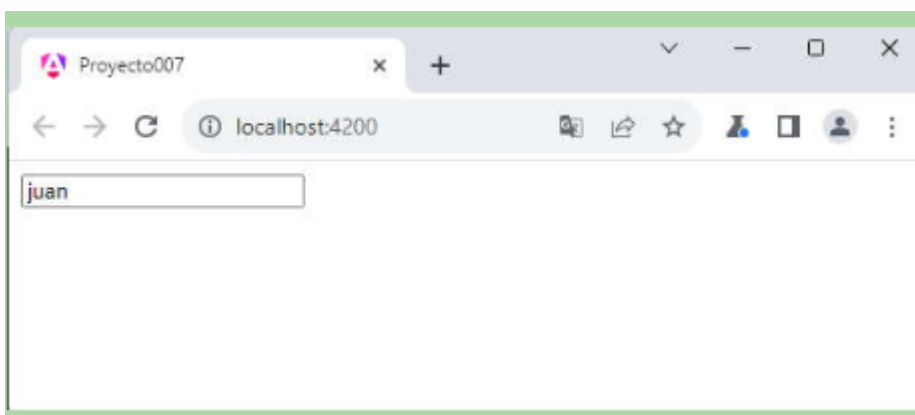
Por ejemplo si en la clase AppComponent tenemos la propiedad 'nombre' con el valor 'juan':

```
nombre='juan';
```

Luego en la vista definimos la directiva ngModel entre corchetes y le asignamos el nombre de la propiedad definida en la clase:

```
<input type="text" [ngModel]="nombre">
```

Cuando arrancamos la aplicación podemos observar que el control input aparece automáticamente con el valor 'juan':



Para utilizar la directiva ngModel debemos importar el módulo 'FormsModule':

```
import { Component } from '@angular/core';
import { RouterOutlet } from '@angular/router';
import { FormsModule } from '@angular/forms';

@Component({
  selector: 'app-root',
  standalone: true,
  imports: [RouterOutlet, FormsModule],
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  nombre = 'juan';
}
```

Lo que hay que tener en cuenta que es un enlace en una única dirección: el valor de la propiedad de la clase se refleja en la interfaz visual. Si el operador cambia el contenido del control 'input' por ejemplo por el nombre 'ana' luego la propiedad 'nombre' de la clase sigue almacenando el valor 'juan'.

Si queremos que el enlace sea en las dos direcciones debemos utilizar la siguiente sintaxis:

```
<input type="text" [(ngModel)]="nombre">
```

Un primer ejemplo muy corto que podemos hacer es crear un proyecto (proyecto007) para que se ingrese el nombre y apellido de una persona y se muestre inmediatamente en la parte inferior.

```
import { Component } from '@angular/core';
import { RouterOutlet } from '@angular/router';
import { FormsModule } from '@angular/forms';

@Component({
  selector: 'app-root',
  imports: [RouterOutlet, FormsModule],
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  nombre = '';
  apellido = '';
}
```

```
}
```

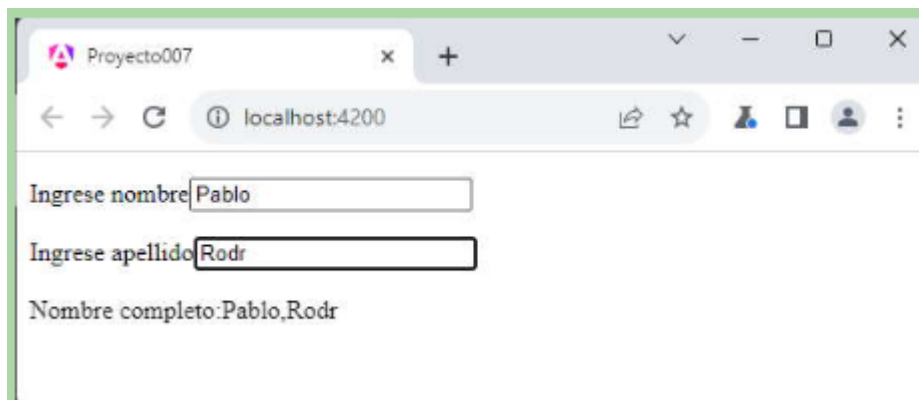
El archivo app.component.html donde definimos las directiva ngModel para cada control es:

```
<div>
  <p>Ingrese nombre<input type="text"
[(ngModel)]="nombre"></p>
  <p>Ingrese apellido<input type="text"
[(ngModel)]="apellido"></p>
  <p>Nombre completo:{{nombre}},{{apellido}}</p>
</div>

<router-outlet />
```

Ejecutemos nuestra aplicación desde la línea de comandos de Node.js.

En el navegador nos muestra como se actualizan las propiedades cada vez que ingresamos un caracter en los controles 'input':



Es importante entender que hay un enlace en ambas direcciones. Si cambiamos el valor mediante el ingreso de datos por teclado en el control input, luego se ven reflejados en forma automática en las propiedades de la clase. De la misma forma si cambiamos el valor de la propiedad en la clase, luego se refleja en forma automática el valor en el formulario.

Administración de artículos

Codigo	Descripcion	Precio	Borrar	Seleccionar
1	papas	10.55	<input data-bbox="842 365 938 398" type="button" value="Borrar?"/>	<input data-bbox="1066 365 1193 398" type="button" value="Seleccionar"/>
2	manzanas	12.1	<input data-bbox="842 427 938 461" type="button" value="Borrar?"/>	<input data-bbox="1066 427 1193 461" type="button" value="Seleccionar"/>
3	melon	52.3	<input data-bbox="842 490 938 524" type="button" value="Borrar?"/>	<input data-bbox="1066 490 1193 524" type="button" value="Seleccionar"/>
4	cebollas	17	<input data-bbox="842 553 938 586" type="button" value="Borrar?"/>	<input data-bbox="1066 553 1193 586" type="button" value="Seleccionar"/>
5	calabaza	20	<input data-bbox="842 616 938 649" type="button" value="Borrar?"/>	<input data-bbox="1066 616 1193 649" type="button" value="Seleccionar"/>

Codigo:

descripcion:

precio: