

VALIDACIÓN

Reactive Forms

```
import { Component, OnInit } from '@angular/core';
import { FormGroup, Validators } from '@angular/forms';

@Component({
  selector: 'app-edit',
  templateUrl: './edit.component.html',
  styleUrls: ['./edit.component.css']
})
export class EditComponent implements OnInit {
  // Atributos
  public usuario: FormGroup;

  constructor(private formBuilder: FormBuilder) {
    this.usuario = this.formBuilder.group({
      nombre: ['', Validators.required],
      apellido: ['', [Validators.required, Validators.minLength(4)]],
      direccion: this.formBuilder.group({
        calle: [''],
        ciudad: ['']
      })
    });
  }

  ngOnInit(): void {
  }
}
```

VALIDACIÓN

A diferencia de la validación en los formularios de plantilla, en los formulario reactivos añadimos los requisitos de validación a continuación de la declaración de parámetro.

```
<form [formGroup]="usuario">
  <label>Nombre:</label>
  <input type="text" formControlName="nombre">
  <div *ngIf="usuario.get('nombre').invalid && usuario.get('nombre').dirty">
    <div *ngIf="usuario.get('nombre').errors.required"> El nombre es obligatorio </div>
  </div>

  <label>Apellido:</label>
  <input type="text" formControlName="apellido">
  <div *ngIf="usuario.get('apellido').invalid && usuario.get('apellido').dirty">
    <div *ngIf="usuario.get('apellido').errors.required"> El apellido es obligatoria </div>
    <div *ngIf="usuario.get('apellido').errors.minLength"> Apellido minimo 4 caracteres </div>
  </div>

  <div formGroupName="direccion">
    <label>Calle:</label>
    <input type="text" formControlName="calle">

    <label>Ciudad:</label>
    <input type="text" formControlName="ciudad">
  </div>

  <button type="submit" [disabled]="!usuario.valid">Crear</button>
</form>
```

VALIDACIÓN

En el HTML será parecido a los formulario de plantilla salvo:

- No es necesario colocar los atributos como *required* o *minLength*.
- Para recuperar el valor del atributo.