Creating reactive forms in Angular is a powerful way to build dynamic, scalable, and testable forms. Here's a **step-by-step guide** to help you master it, along with **video tutorials** that walk you through the process visually.

🧠 Step-by-Step: How to Create Reactive Forms in Angular

1. **Import ReactiveFormsModule**

In your AppModule or app.config.ts, import the module:

import { ReactiveFormsModule } from '@angular/forms';

@NgModule({

imports: [ReactiveFormsModule],

})

export class AppModule {}

Or in standalone setup:

import { provideReactiveForms } from '@angular/forms';

export const appConfig: ApplicationConfig = {

providers: [provideReactiveForms()]

};

2. **Create a Form in Component Class**

Use FormGroup and FormControl to define your form

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

import { FormGroup, FormControl, Validators } from '@angular/forms';

@Component({

selector: 'app-user-form',

templateUrl: './user-form.component.html',

})

export class UserFormComponent {

userForm = new FormGroup({

name: new FormControl('', Validators.required),

email: new FormControl('', [Validators.required, Validators.email]),

age: new FormControl('', Validators.min(18)),

});

onSubmit() {

console.log(this.userForm.value);

}

}

3. **Bind Form in Template**

Use [formGroup] and formControlName:

<form [formGroup]="userForm" (ngSubmit)="onSubmit()">

<label>Name:</label>

<input formControlName="name" />

<label>Email:</label>

<input formControlName="email" />

<label>Age:</label>

<input formControlName="age" type="number" />

<button type="submit">Submit</button>

</form>

4. **Add Validation Feedback**

<div \*ngIf="userForm.get('name')?.invalid && userForm.get('name')?.touched">

Name is required.

</div>