**1 – Instalação do ReactiveFormsModule**

Antes de começar, você precisa importar o módulo **ReactiveFormsModule** no seu projeto.  
Esse modulo possui todas as diretivas de criação de formulários do tipo Reactive Forms. É necessário efetuar o import antes de qualquer componente que use um formulário.  
Para importar, é só incluir a linha abaixo no arquivo **app.module.ts**.

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

O arquivo **app.module.ts** deve ficar assim:

import { BrowserModule } from '@angular/platform-browser';

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

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

import { AppComponent } from './app.component';

@NgModule({

declarations: [

AppComponent

],

imports: [

BrowserModule,

ReactiveFormsModule

],

providers: [],

bootstrap: [AppComponent]

})

export class AppModule { }

## 2 – Criar a classe que será o modelo do formulário

Para esse exemplo, será criado uma classe de cliente com os campos abaixo.  
Para criar uma classe é só rodar o comando abaixo.

ng g class clientes/shared/cliente

O arquivo **cliente.ts** deve ficar assim:

export class Cliente {

    nome: string = '';

    tipo: number = 1;

    genero: number = 1;

    dataNascimento: string= '';

    observacao: string = '';

    inativo: boolean = false;

}

## 3 – Criar o componente com o formulário

O formulário no Angular é um formulário comum com a adição do formControlName para fazer o binding entre o controle na tela e o objeto de formulário que você está usando.

**Criar o componente**  
Para criar o componente que terá o formulário, é só rodar o comando abaixo

ng g component clientes/formulario

A principio o arquivo **formulario.component.html** deve ficar assim:

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

  <div class="form-group">

    <label for="nome">Nome</label>

    <input type="text" class="form-control" name="name" id="nome" formControlName="nome">

  </div>

  <div class="form-group">

    <label for="tipo">Tipo</label>

    <select class="form-control" name="tipo" id="tipo" formControlName="tipo">

      <option value="1">Pessoa física</option>

      <option value="2">Pessoa jurídica</option>

    </select>

  </div>

  <div class="form-group">

    <label>Gênero</label>

    <div class="form-check">

      <input class="form-check-input" type="radio" name="genero" id="masculino" [value]="1" formControlName="genero">

      <label class="form-check-label" for="masculino">Masculino</label>

    </div>

    <div class="form-check">

      <input class="form-check-input" type="radio" name="genero" id="feminino" [value]="2" formControlName="genero">

      <label class="form-check-label" for="feminino">Feminino</label>

    </div>

  </div>

  <div class="form-group">

    <label for="dataNascimento">Data de nascimento</label>

    <input type="date" class="form-control" name="dataNascimento" id="dataNascimento" formControlName="dataNascimento">

  </div>

  <div class="form-group">

    <label for="observacao">Observação</label>

    <textarea class="form-control" rows="3" name="observacao" id="observacao" formControlName="observacao"></textarea>

  </div>

  <div class="form-check">

    <input class="form-check-input" type="checkbox" name="inativo" id="inativo" formControlName="inativo">

    <label class="form-check-label" for="inativo">Inativo</label>

  </div>

  <input type="submit" class="btn btn-primary mt-4" value="Salvar">

  <a href="#" class="btn btn-secondary mt-4 ml-2">Cancelar</a>

</form>

NO FORMULARIO.TS

import { Cliente } from '../shared/cliente';

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

export class FormularioComponent implements OnInit {

  formCliente!: FormGroup;

  constructor() { }

ngOnInit(): void {

    this.createForm(new Cliente());

  }

 createForm(cliente: Cliente) {

    this.formCliente = new FormGroup({

      nome: new FormControl(cliente.nome),

      tipo: new FormControl(cliente.tipo),

      genero: new FormControl(cliente.genero),

      dataNascimento: new FormControl(cliente.dataNascimento),

      observacao: new FormControl(cliente.observacao),

      inativo: new FormControl(cliente.inativo)

    })

  }

## 5 – Fazer o formulário submter

Para fazer o formulário submeter é necessário criar um método no arquivo **formulario.component.ts** e adicionar a chamada desse método no evento (ngSubmit) do formulário.

onSubmit() {

   // aqui você pode implementar a logica para fazer seu formulário salvar

   console.log(this.formCliente.value);

    // chamando a função createForm para limpar os campos na tela

    this.createForm(new Cliente());

  }