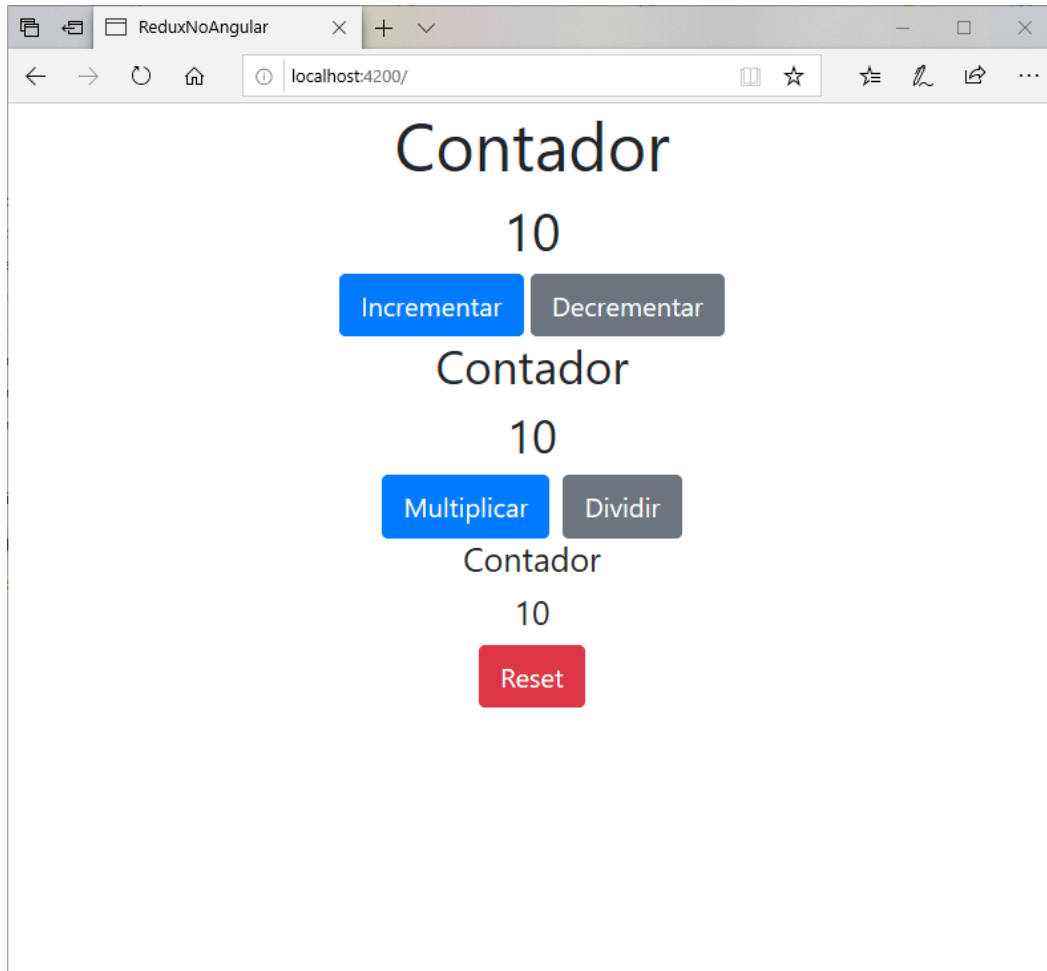




@Input()

@Output()



ng new decoradores-input-output

ng g c hijo

ng g c nieto

/src/app/app-component.html

```
<div class="container">

  <div class="row">
    <div class="col-12 text-center">
      <h1>Contador</h1>
      <h2>{{ contador }}</h2>
    </div>
  </div>
  <div class="row">
    <div class="col text-center">
      <button class="btn btn-primary" (click)="incrementar()" >Incrementar</button>
      <button class="btn btn-secondary ml-1" (click)="decrementar()" >Degrementar</button>
    </div>
  </div>
  <app-hijo [contador]="contador" (cambioContador)="contador=$event" ></app-hijo>
</div>

<router-outlet></router-outlet>
```



/src/app/app-component.ts

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

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  contador: number;

  constructor() {
    this.contador = 10;
  }

  incrementar(){
    this.contador++;
  }

  decrementar(){
    this.contador--;
  }
}
```

## /src/app/hijo/hijo-component.html

```
<div class="row">
  <div class="col text-center">
    <h3>Contador</h3>
    <h3>{{contador}}</h3>
  </div>
</div>
<div class="row">
  <div class="col text-center">
    <button class="btn btn-primary" (click)="multiplicar()">Multiplicar</button>
    <button class="btn btn-secondary ml-2" (click)="dividir()">Dividir</button>
  </div>
</div>

<app-nieto [contador]="contador" (contadorCambio)="resetNieto($event)"></app-
nieto>
```



### /src/app/hijo/hijo-component.ts

```
import { Component, OnInit, Input, Output ,EventEmitter } from '@angular/core';

@Component({
  selector: 'app-hijo',
  templateUrl: './hijo.component.html',
  styleUrls: ['./hijo.component.css']
})
export class HijoComponent implements OnInit {

  @Input() contador;
  @Output() cambioContador = new EventEmitter<number>();

  constructor() { }

  ngOnInit() {
  }

  multiplicar(){
    this.contador*=2;
    this.cambioContador.emit(this.contador);
  }

  dividir(){
    this.contador/=2;
    this.cambioContador.emit(this.contador);
  }

  resetNieto(nuevoContador){
    this.contador = nuevoContador;
    this.cambioContador.emit(this.contador);
  }
}
```

/src/app/nieto/nieto-compoment.html

```
<div class="row">
  <div class="col text-center">
    <h5>Contador</h5>
    <h5>{{contador}}</h5>
  </div>
</div>
<div class="row">
  <div class="col text-center">
    <button class="btn btn-danger" (click)="reset()">Reset</button>
  </div>
</div>
```



### /src/app/nieto/nieto-component.ts

```
import { Component, OnInit, Input, Output, EventEmitter } from '@angular/core';

@Component({
  selector: 'app-nieto',
  templateUrl: './nieto.component.html',
  styleUrls: ['./nieto.component.css']
})
export class NietoComponent implements OnInit {

  @Input() contador;
  @Output() contadorCambio = new EventEmitter<number>();

  constructor() { }

  ngOnInit() {
  }

  reset(){
    this.contador = 0;
    this.contadorCambio.emit(this.contador);
  }
}
```