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

import { NgForm} from '@angular/forms'

@Component({

  selector: 'app-calculator',

  templateUrl: './calculator.component.html',

  styleUrls: ['./calculator.component.css']

})

export class CalculatorComponent implements OnInit {

  val7:string="7"

  val1:string=""

  dis:string=""

  len

  i:number=0

  sum:number=0

  fval:string

  constructor() { }

  ngOnInit(): void {

  }

  fun(value:string){

    // this.val1+=value;

    // console.log(value);

  }

  cal(value:NgForm){

    console.log(value);

  }

  funbtn(val:string){

    if (val=='+'){

      this.len=this.val1.length;

      // console.log(this.len);

      // for( this.i=0; this.i<this.len; this.i++){

      //   this.fval=this.val1.charAt(this.i);

      // }

      var n=parseInt(this.val1)

      console.log(n);

      this.val1="";

      // this.dis="none";

    }

    else{

    this.val1+=val;

    // this.dis=""

    console.log(val);

    }

  }

}

<!-- <p>

    {{val7}}

</p> -->

<p [ngStyle]="{'display':dis}">{{val1}}</p>

<div>

<!-- <form #f="ngForm" (ngSubmit)="cal(f)"> -->

    <input type="text" #numm7 name="num7" (click)="fun(numm7)" [(ngModel)]="val7" > &nbsp; <button>8</button> &nbsp; <button>9</button> <br>

    <button>4</button> &nbsp; <button>5</button> &nbsp; <button>6</button> <br>

    <button  (click)="funbtn('1')">1</button> &nbsp; <button (click)="funbtn('2')">2</button> &nbsp; <button>3</button> <br>

    <button>0</button> &nbsp; <button>=</button> &nbsp; <button (click)="funbtn('+')">+</button>

<!-- </form> -->

<!-- <input type="text" > -->

</div>

<div>

    <input type="text">

</div>

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

import { NgForm} from '@angular/forms'

@Component({

  selector: 'app-calculator',

  templateUrl: './calculator.component.html',

  styleUrls: ['./calculator.component.css']

})

export class CalculatorComponent implements OnInit {

  val7:string="7"

  val1:string=""

  dis:string=""

  len

  i:number=0

  sum:number=0

  fval:string

  n1:number

  n2:number

  last

  ans

  constructor() { }

  ngOnInit(): void {

  }

  fun(value:string){

    // this.val1+=value;

    // console.log(value);

  }

  cal(value:NgForm){

    console.log(value);

  }

  funbtn(val:string){

    if (val=='+' || val=='-' || val=='\*' || val=='/'){

      //this.len=this.val1.length;

      // console.log(this.len);

      // for( this.i=0; this.i<this.len; this.i++){

      //   this.fval=this.val1.charAt(this.i);

      // }

      this.last=val

      console.log(this.last)

      this.n1=parseInt(this.val1)

      //this.n1=this.n1+5

      console.log(this.n1);

      this.val1="";

    }

    else if(val=='='){

      this.n2=parseInt(this.val1)

      // this.n1=this.n1+this.n2

      console.log(this.n2);

      switch(this.last){

        case  '+' :

          {

            this.ans=this.n1+this.n2

          }

          break;

          case '-':

            {

              this.ans=this.n1-this.n2

            }

          break;

          case '\*':

            {

              this.ans=this.n1\*this.n2

            }

          break;

          case '/':

            {

              this.ans=this.n1/this.n2

            }

          break;

      }

      this.val1="";

    }

    else{

    this.val1+=val;

    // this.dis=""

    console.log(val);

    }

  }

}

<!-- <p>

    {{val7}}

</p> -->

<p [ngStyle]="{'display':dis}">{{val1}}</p>

<p>{{ans}}</p>

<div>

<!-- <form #f="ngForm" (ngSubmit)="cal(f)"> -->

    <button (click)="funbtn('7')">7</button> &nbsp; <button (click)="funbtn('8')">8</button> &nbsp; <button (click)="funbtn('9')">9</button> <br>

    <button (click)="funbtn('4')">4</button> &nbsp; <button (click)="funbtn('5')">5</button> &nbsp; <button (click)="funbtn('6')">6</button> <br>

    <button  (click)="funbtn('1')">1</button> &nbsp; <button (click)="funbtn('2')">2</button> &nbsp; <button (click)="funbtn('3')">3</button> <br>

    <button>0</button> &nbsp; <button (click)="funbtn('=')">=</button> &nbsp; <button (click)="funbtn('+')">+</button>

    &nbsp; <br><button (click)="funbtn('-')">-</button> &nbsp; <button (click)="funbtn('\*')">\*</button> &nbsp; <button (click)="funbtn('/')">/</button>

<!-- </form> -->

<!-- <input type="text" > -->

</div>

<!-- <p>{{ans}}</p> -->

<div>

    <input type="text">

</div>

Improve

value:string=""

  len:number;

  n1:number;

  n2:number;

  last:string;

  ans:number;

  constructor() { }

  ngOnInit(): void {

  }

  cal(value:NgForm){

    console.log(value);

  }

  funbtn(val:string){

    if (val=='+' || val=='-' || val=='\*' || val=='/'){

      this.last=val

      console.log(this.last)

      this.n1=parseInt(this.value)

      console.log(this.n1);

      // this.val1=val;

    }

    else if(val=='='){

      this.n2=parseInt(this.value)

      console.log(this.n2);

      switch(this.last){

        case  '+' :

          {

            this.ans=this.n1+this.n2

            this.value=this.ans.toString();

          }

          break;

          case '-':

            {

              this.ans=this.n1-this.n2

              this.value=this.ans.toString();

            }

          break;

          case '\*':

            {

              this.ans=this.n1\*this.n2

              this.value=this.ans.toString();

            }

          break;

          case '/':

            {

              this.ans=this.n1/this.n2

              this.value=this.ans.toString();

            }

          break;

      }

    }

    else {

      if(this.last!=null)

      this.value=""

    this.value+=val;

    console.log(val);

    }

  }

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

import { NgForm} from '@angular/forms'

@Component({

  selector: 'app-calculator',

  templateUrl: './calculator.component.html',

  styleUrls: ['./calculator.component.css']

})

export class CalculatorComponent implements OnInit {

  value:string=""

  length:number;

  n1:number=0;

  n2:number=0;

  lastValue:string="";

  answer:number;

  count:number;

  constructor() { }

  ngOnInit(): void {

  }

  funbtn(val:string){

    if(this.value=="Error" || this.value=="NaN")

    {this.value="Error"

    if(val=='')

    this.value=""

    }

    else if (val=='+' || val=='-' || val=='\*' || val=='/'){

      if(this.value!=null){

      this.lastValue=val

      //console.log(this.lastValue)

      this.n1=parseFloat(this.value)

      this.value=""

      console.log("Op"+this.n1);

      ;

    }

  }

    else if(val=='='){

      if(this.value!=null){

      if(this.count==0)

      this.n2=parseFloat(this.value)

      console.log("N2"+this.n2);

      console.log("Last"+this.lastValue)

      switch(this.lastValue){

        case  '+' :

          {

            if(this.count!=0)

            {this.answer=this.answer+this.n2;

            this.value=this.answer.toString();

            console.log("Answer"+this.answer)

            }

            else{

            this.answer=this.n1+this.n2

            this.value=this.answer.toString();

            if (this.value=="NaN")

              { this.value="Error"}

            this.count++;}

          }

          break;

          case '-':

            {

              if(this.count!=0)

              {this.answer=this.answer-this.n2;

              this.value=this.answer.toString();

              console.log("Answer"+this.answer)

              }

              else{

              this.answer=this.n1-this.n2

              this.value=this.answer.toString();

              if (this.value=="NaN")

              { this.value="Error"}

              this.count++;

              }

            }

          break;

          case '\*':

            {

              if(this.count!=0)

            {this.answer=this.answer\*this.n2;

            this.value=this.answer.toString();

            if(this.value=="Infinity" || this.value=="NaN")

            {this.value="Error";

            //this.answer=0;

            this.n2=0;

            this.n1=0;

            this.lastValue="";

            //this.value="";

            }

            console.log("Answer"+this.answer)

            }

            else{

              this.answer=this.n1\*this.n2

              this.value=this.answer.toString();

              if (this.value=="NaN")

              { this.value="Error"}

              this.count++;

            }

            }

          break;

          case '/':

            {

              if(this.count!=0)

            {this.answer=this.answer/this.n2;

            this.value=this.answer.toString();

            console.log("Answer"+this.answer)

            }

            else{

              this.answer=this.n1/this.n2

              this.value=this.answer.toString();

              if (this.value=="NaN")

              { this.value="Error"}

              if(this.value=="Infinity"){

                this.value="Error";

              }

              this.count++;

            }

              console.log("Equal"+this.value)

            }

          break;

      }

    }

  }

    else if (val==''){

      this.value="";

     console.log("Empty"+this.value)

    }

    else {

    this.value+=val;

    this.count=0;

    console.log("Val"+val);

    }

  }

}

<div [ngClass]="'Screen'">

    <p>{{value}}</p>

</div>

<br>

<div>

    <button (click)="funbtn('7')">7</button> &nbsp; <button (click)="funbtn('8')">8</button> &nbsp; <button (click)="funbtn('9')">9</button> &nbsp; <button (click)="funbtn('\*')">\*</button> <br>

    <button (click)="funbtn('4')">4</button> &nbsp; <button (click)="funbtn('5')">5</button> &nbsp; <button (click)="funbtn('6')">6</button> &nbsp; <button (click)="funbtn('/')">/</button> <br>

    <button (click)="funbtn('1')">1</button> &nbsp; <button (click)="funbtn('2')">2</button> &nbsp; <button (click)="funbtn('3')">3</button> &nbsp; <button (click)="funbtn('-')">-</button> <br>

    <button (click)="funbtn('=')">=</button> &nbsp; <button (click)="funbtn('0')">0</button> &nbsp; <button (click)="funbtn('+')">+</button> &nbsp; <button (click)="funbtn('')">C</button>  <br>

</div>

This is an