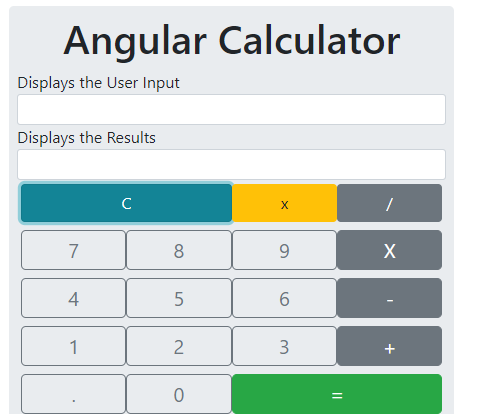
Exercises on Angular

1. Design a form as shown in the picture below



* + Create an Angular application named “calculator-app”
  + You have to create a folder ‘calculator’ in app folder under src folder. And you need to write all the programs in this folder.
  + Include bootstrap library in this application

Calculator.component.ts

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

@Component({

    selector:"calculator",

    templateUrl:"./calculator.component.html"

})

export class CalculatorComponent {

    input: string = '';

    result: string = '';

    pressNum(num: string) {

        //Do Not Allow . more than once

        if (num == ".") {

          if (this.input != "") {

            const lastNum = this.getLastOperand()

            console.log(lastNum.lastIndexOf("."))

            if (lastNum.lastIndexOf(".") >= 0) return;

          }

        }

        //Do Not Allow 0 at beginning.

        //Javascript will throw Octal literals are not allowed in strict mode.

        if (num == "0") {

          if (this.input == "") {

            return;

          }

          const PrevKey = this.input[this.input.length - 1];

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

            return;

          }

        }

        this.input = this.input + num

        this.calcAnswer();

      }

      getLastOperand() {

        let pos: number;

        console.log(this.input)

        pos = this.input.toString().lastIndexOf("+")

        if (this.input.toString().lastIndexOf("-") > pos) pos = this.input.lastIndexOf("-")

        if (this.input.toString().lastIndexOf("\*") > pos) pos = this.input.lastIndexOf("\*")

        if (this.input.toString().lastIndexOf("/") > pos) pos = this.input.lastIndexOf("/")

        console.log('Last ' + this.input.substr(pos + 1))

        return this.input.substr(pos + 1)

      }

      pressOperator(op: string) {

        //Do not allow operators more than once

        const lastKey = this.input[this.input.length - 1];

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

          return;

        }

        this.input = this.input + op

        this.calcAnswer();

      }

      clear() {

        if (this.input != "") {

          this.input = this.input.substr(0, this.input.length - 1)

        }

      }

      allClear() {

        this.result = '';

        this.input = '';

      }

      calcAnswer() {

        let formula = this.input;

        let lastKey = formula[formula.length - 1];

        if (lastKey === '.') {

          formula = formula.substr(0, formula.length - 1);

        }

        lastKey = formula[formula.length - 1];

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

          formula = formula.substr(0, formula.length - 1);

        }

        console.log("Formula " + formula);

        this.result = eval(formula);

      }

      getAnswer() {

        this.calcAnswer();

        this.input = this.result;

        if (this.input == "0") this.input = "";

      }

}

Calculator.component.html

<div class="container-fluid">

    <div class="jumbotron col-sm-4 p-2 m-0 bg-inverse">

      <h1 class="text-center">Angular Calculator</h1>

      Displays the User Input

      <div class="input-group input-group-sm col-sm-12 m-0 p-0">

        <div class="col-sm-12 form-control text-lg-right" type="text">

            {{input}}

        </div>

      </div>

      Displays the Results

      <div class="input-group input-group-sm col-sm-12 m-0 p-0">

        <div class="form-control text-sm-right" type="text">

            {{result}}

        </div>

      </div>

      <div class="col-sm-12 p-1 m-0">

        <button class="btn btn-info col-sm-6" type="button"

            (click)="allClear()">C</button>

        <button class="btn btn-warning col-sm-3" type="button"

            (click)="clear()">x</button>

        <button class="btn btn-secondary col-sm-3" type="button"

            (click)="pressOperator('/')">/</button>

      </div>

      <div class="col-sm-12 p-1 m-0">

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('7')">7</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('8')">8</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('9')">9</button>

        <button class="btn btn-lg btn-secondary col-sm-3 p-1"

            type="button" (click)="pressOperator('\*')">X</button>

      </div>

      <div class="col-sm-12 p-1 m-0">

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('4')">4</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('5')">5</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('6')">6</button>

        <button class="btn btn-lg btn-secondary col-sm-3 p-1"

        type="button" (click)="pressOperator('-')">-</button>

      </div>

      <div class="col-sm-12 p-1 m-0">

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('1')">1</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('2')">2</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('3')">3</button>

        <button class="btn btn-lg btn-secondary col-sm-3 p-1"

            type="button" (click)="pressOperator('+')">+</button>

      </div>

      <div class="col-sm-12 p-1 m-0">

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('.')">.</button>

        <button class="btn btn-lg btn-outline-secondary col-sm-3 p-1"

            type="button" (click)="pressNum('0')">0</button>

        <button class="btn btn-lg btn-success col-sm-6 p-1"

            type="button" (click)="getAnswer()">=</button>

      </div>

    </div>

Add bootstrap in index.html

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>My Beautiful Calculator</title>

  <base href="/">

  <meta name="viewport" content="width=device-width, initial-scale=1">

  <link rel="icon" type="image/x-icon" href="favicon.ico">

<link

    rel="stylesheet"

    href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min.css"

    integrity="sha384-9aIt2nRpC12Uk9gS9baDl411NQApFmC26EwAOH8WgZl5MYYxFfc+NcPb1dKGj7Sk"

    crossorigin="anonymous">

</head>

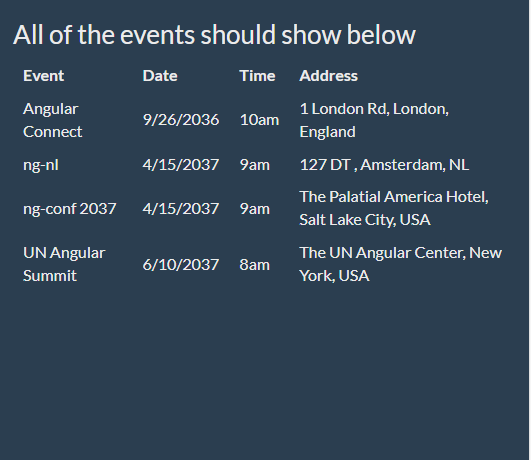
<body>

  <app-root></app-root>

</body>

</html>

1. You need to get an output shown in the below picture



* + Create an Angular application named “event-app”
  + You have to create a folder ‘event’ in app folder under src folder. And you need to write all the programs in this folder.
  + Include bootstrap library in this application
  + Create set of events in the class as class variable as json

 const events = [

    {name:'Angular Connect', date: '9/26/2036', time: '10am', location: {address: '1 London Rd', city: 'London', country: 'England'}},

    {name:'ng-nl', date: '4/15/2037', time: '9am', location: {address: '127 DT ', city: 'Amsterdam', country: 'NL'}},

    {name:'ng-conf 2037', date: '4/15/2037', time: '9am', location: {address: 'The Palatial America Hotel', city: 'Salt Lake City', country: 'USA'}},

    {name:'UN Angular Summit', date: '6/10/2037', time: '8am', location: {address: 'The UN Angular Center', city: 'New York', country: 'USA'}},

  ]

event.component.ts

import { Component } from '@angular/core'

@Component({

  selector: 'event-details',

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

  styles: [`

    th, td { font-size: 16px; padding:5px 10px;}

  `]

})

export class EventsListComponent {

  const events = [

    {name:'Angular Connect', date: '9/26/2036', time: '10am', location: {address: '1 London Rd', city: 'London', country: 'England'}},

    {name:'ng-nl', date: '4/15/2037', time: '9am', location: {address: '127 DT ', city: 'Amsterdam', country: 'NL'}},

    {name:'ng-conf 2037', date: '4/15/2037', time: '9am', location: {address: 'The Palatial America Hotel', city: 'Salt Lake City', country: 'USA'}},

    {name:'UN Angular Summit', date: '6/10/2037', time: '8am', location: {address: 'The UN Angular Center', city: 'New York', country: 'USA'}},

  ]

}

event.component.html

<div>

  <h3>All of the events should show below</h3>

  <table>

    <thead>

      <tr>

        <th>Event</th>

        <th>Date</th>

        <th>Time</th>

        <th>Address</th>

      </tr>

    </thead>

    <tbody>

      <tr \*ngFor="let event of events">

        <td>{{event.name}}</td>

        <td>{{event.date}} </td>

        <td>{{event.time}}</td>

        <td>{{event.location.address}}, {{event.location.city}}, {{event.location.country}}</td>

      </tr>

    </tbody>

  </table>

</div>

1. Xxxx
2. Xxxx
3. Ddd
4. Ffff
5. fdd