**Scripting Language Lab**

Name: Utkarsh Prakash Srivastava

Sec: B

Reg No.: 201900232

Roll No.: 25

**CODE OF ALL COMPONENTS**

1.*calculator.component.html*

<div class="card">

  <div class="card-header text-center">CALCULATOR</div>

  <div class="card-body">

    <div class="form-group d-flex flex-row">

      <input [(ngModel)]='num1' type="number" name="num1" class="form-control">

      <input [(ngModel)]='num2' type="number" name="num2" class="form-control">

    </div>

    <div class="d-flex flex-row justify-content-between">

      <button class="btn btn-warning" (click)="addition()">SUM</button>

      <button class="btn btn-warning" (click)="subtraction()">SUBTRACTION</button>

      <button class="btn btn-warning" (click)="multiplication()">MULTIPLICATION</button>

      <button class="btn btn-warning" (click)="division()">DIVISION</button>

      <button class="btn btn-warning" (click)="factorial()">FACTORIAL</button>

      <button class="btn btn-warning" (click)="isprime()">CHECK PRIME</button>

    </div>

  </div>

  <div class="card-footer text-center">

    <p> <b>NOTE : For "FACTORIAL" and "CHECK PRIME" enter only in 1st box.</b></p>

    <h4> The Result is {{result}} </h4>

  </div>

</div>

2.*calculator.component.ts*

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

@Component({

  selector: 'app-calculator',

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

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

})

export class CalculatorComponent {

  public num1!: number;

  public num2!: number;

  public result!: String;

addition()

{

  var ans:number = 0;

  ans= this.num1 + this.num2;

  this.result = String(ans);

}

subtraction()

{

  var ans:number = 0;

  ans = this.num1 - this.num2;

  this.result = String(ans);

}

multiplication()

{

  var ans:number = 0;

  ans = this.num1 \* this.num2;

  this.result = String(ans);

}

division()

{

  var ans:number = 0;

  ans = (this.num1) / (this.num2);

  this.result = String(ans);

}

factorial()

{

  var ans:number = 0;

  var fact:number = 1;

  while(this.num1 >=1) {

    fact = fact \* this.num1;

    this.num1--;

 }

 ans = fact;

 this.result = String(ans);

}

isprime()

{

  var temp:number = this.num1;

  var ans:number = 1;

  for(var i = 2; i < temp; i++)

    if(temp % i === 0)

      ans = 0;

  if( ans === 0)

  {

    this.result = " NOT PRIME"

  }

  else{

    this.result = "PRIME"

  }

}

}

*3.app-routing.module.ts*

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

import { RouterModule, Routes } from '@angular/router';

const routes: Routes = [];

@NgModule({

  imports: [RouterModule.forRoot(routes)],

  exports: [RouterModule]

})

export class AppRoutingModule { }

*4.app.component.html*

<app-calculator></app-calculator>

*5.app.component.spec.ts*

import { TestBed } from '@angular/core/testing';

import { RouterTestingModule } from '@angular/router/testing';

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

describe('AppComponent', () => {

  beforeEach(async () => {

    await TestBed.configureTestingModule({

      imports: [

        RouterTestingModule

      ],

      declarations: [

        AppComponent

      ],

    }).compileComponents();

  });

  it('should create the app', () => {

    const fixture = TestBed.createComponent(AppComponent);

    const app = fixture.componentInstance;

    expect(app).toBeTruthy();

  });

  it(`should have as title 'CalculatorApp'`, () => {

    const fixture = TestBed.createComponent(AppComponent);

    const app = fixture.componentInstance;

    expect(app.title).toEqual('CalculatorApp');

  });

  it('should render title', () => {

    const fixture = TestBed.createComponent(AppComponent);

    fixture.detectChanges();

    const compiled = fixture.nativeElement as HTMLElement;

    expect(compiled.querySelector('.content span')?.textContent).toContain('CalculatorApp app is running!');

  });

});

*6. app.component.ts*

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

@Component({

  selector: 'app-root',

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

  styleUrls: ['./app.component.css']

})

export class AppComponent {

  title = 'CalculatorApp';

}

*7. app.module.ts*

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

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

import { AppRoutingModule } from './app-routing.module';

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

import { CalculatorComponent } from './calculator/calculator.component';

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

@NgModule({

  declarations: [

    AppComponent,

    CalculatorComponent

  ],

  imports: [

    BrowserModule,

    AppRoutingModule,

    FormsModule

  ],

  providers: [],

  bootstrap: [AppComponent]

})

export class AppModule { }

*8.index.html*

<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8">

  <title>CalculatorApp</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/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">

</head>

<body>

  <app-root></app-root>

</body>

</html>

**WEB APP**

1. *INITIAL LOOK*

Graphical user interface, text, application, email, website

Description automatically generated

1. *Case1 : Addition*

Graphical user interface, text, application, email, website

Description automatically generated

1. *CASE 2: SUBTRACTION*

Graphical user interface, text, application, email, website

Description automatically generated

1. *CASE 3: MULTIPLICATION*

Graphical user interface, text, application, email, website

Description automatically generated

1. *CASE 4 : DIVISION*

Graphical user interface, application

Description automatically generated

1. *CASE 5: FACTORIAL*

Graphical user interface, text, application, email, website

Description automatically generated

1. *CASE 6.a: CHECK PRIME(is prime)*

Graphical user interface, text, application, email, website

Description automatically generated

1. *CASE 6.b: CHECK PRIME(is not prime)*

Graphical user interface, application, email, website

Description automatically generated