**Ex: 1 An application that uses GUI components, Fonts, Colours**

**Date: 18/08/2022**

**Aim:**

To create a mobile application that uses GUI components, fonts, and colours.

**Code:**

import 'package:flutter/material.dart';

void main() {

runApp(MaterialApp(

home: Home(),

));

}

class Home extends StatelessWidget {

const Home({Key? key}) : super(key: key);

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: Text("Hello World"),

centerTitle: true,

backgroundColor: Color.fromARGB(255, 34, 126, 255)),

body: Center(

child: Text(

"Hello World",

style: TextStyle(

fontSize: 45.0,

fontWeight: FontWeight.bold,

letterSpacing: 2.0,

color: Colors.blueGrey[600],

fontFamily: 'Arial',

),

),

),

floatingActionButton: FloatingActionButton(

onPressed: () {},

child: Text("+"),

backgroundColor: Color.fromARGB(255, 34, 126, 255),

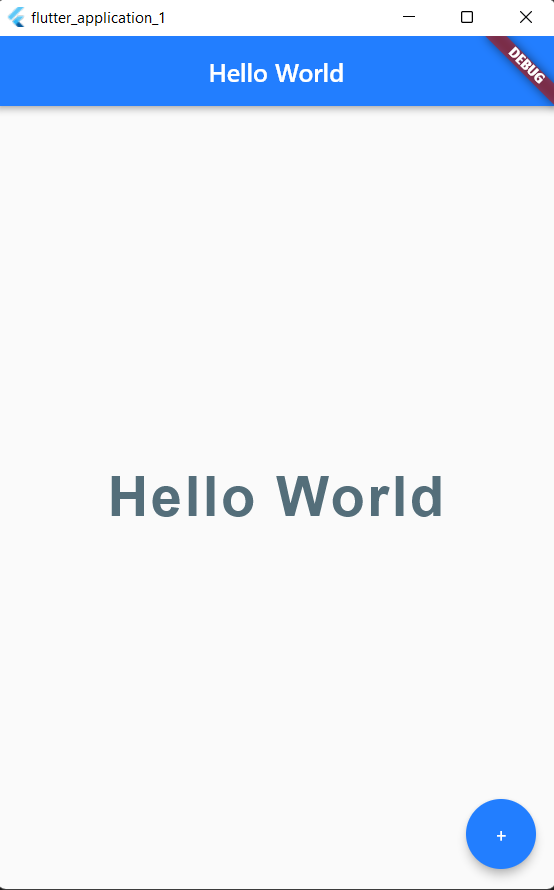
),

);

}

}

**Output:**

****

**Result:**

A mobile application which uses GUI components, fonts, and colours has been implemented successfully

**Ex: 2 An application that uses Layout Managers and Event Listeners**

**Date: 25/08/2022**

**Aim:**

To create a mobile application that uses Layout Managers and Event Listeners

**Code:**

import 'package:flutter/material.dart';

void main() {

runApp(const MaterialApp(

home: Home(),

));

}

class Home extends StatefulWidget {

const Home({Key? key}) : super(key: key);

@override

State<Home> createState() => \_HomeState();

}

class \_HomeState extends State<Home> {

int projects = 0;

@override

Widget build(BuildContext context) {

return Scaffold(

backgroundColor: Color.fromARGB(255, 223, 223, 225),

appBar: AppBar(

title: Text("Sample layout"),

backgroundColor: Colors.black12,

centerTitle: true,

elevation: 0.0,

),

body: Padding(

padding: EdgeInsets.fromLTRB(30.0, 40.0, 30.0, 0.0),

child: Column(

crossAxisAlignment: CrossAxisAlignment.start,

children: <Widget>[

Center(

child: CircleAvatar(

backgroundImage: AssetImage('assets/flutter.png'),

radius: 50.0,

),

),

SizedBox(

height: 20.0,

),

Text(

"Sample layout",

style: TextStyle(

color: Colors.black,

letterSpacing: 2.0,

),

),

SizedBox(

height: 10.0,

),

Text(

"$projects",

style: TextStyle(

color: Colors.blue,

letterSpacing: 2.0,

fontSize: 28.0,

fontWeight: FontWeight.bold,

),

),

SizedBox(

height: 20.0,

),

ElevatedButton(

onPressed: () {

setState(() {

projects++;

});

},

onLongPress: () {

setState(() {

projects \*= 2;

});

},

child: Icon(

Icons.add,

),

style: ElevatedButton.styleFrom(

primary: Colors.green,

),

),

SizedBox(

height: 20.0,

),

ElevatedButton(

onPressed: () {

setState(() {

if (projects > 0) projects--;

});

},

onLongPress: () {

setState(() {

if (projects > 0) projects ~/= 2;

});

},

child: Icon(

Icons.remove,

),

style: ElevatedButton.styleFrom(

primary: Colors.deepOrange,

),

),

SizedBox(

height: 20.0,

),

Row(

children: [

SizedBox(

width: 20.0,

),

],

)

],

),

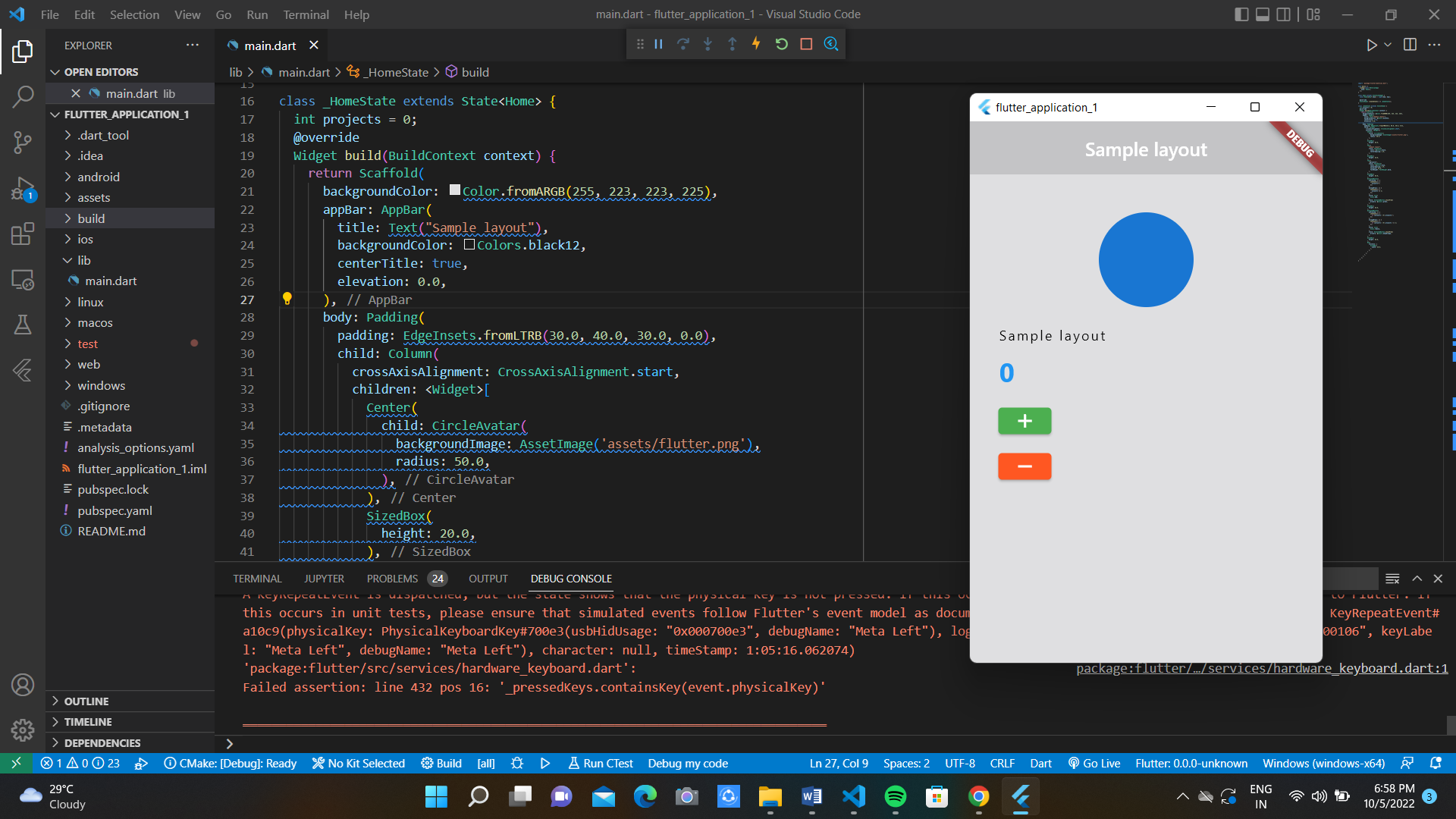
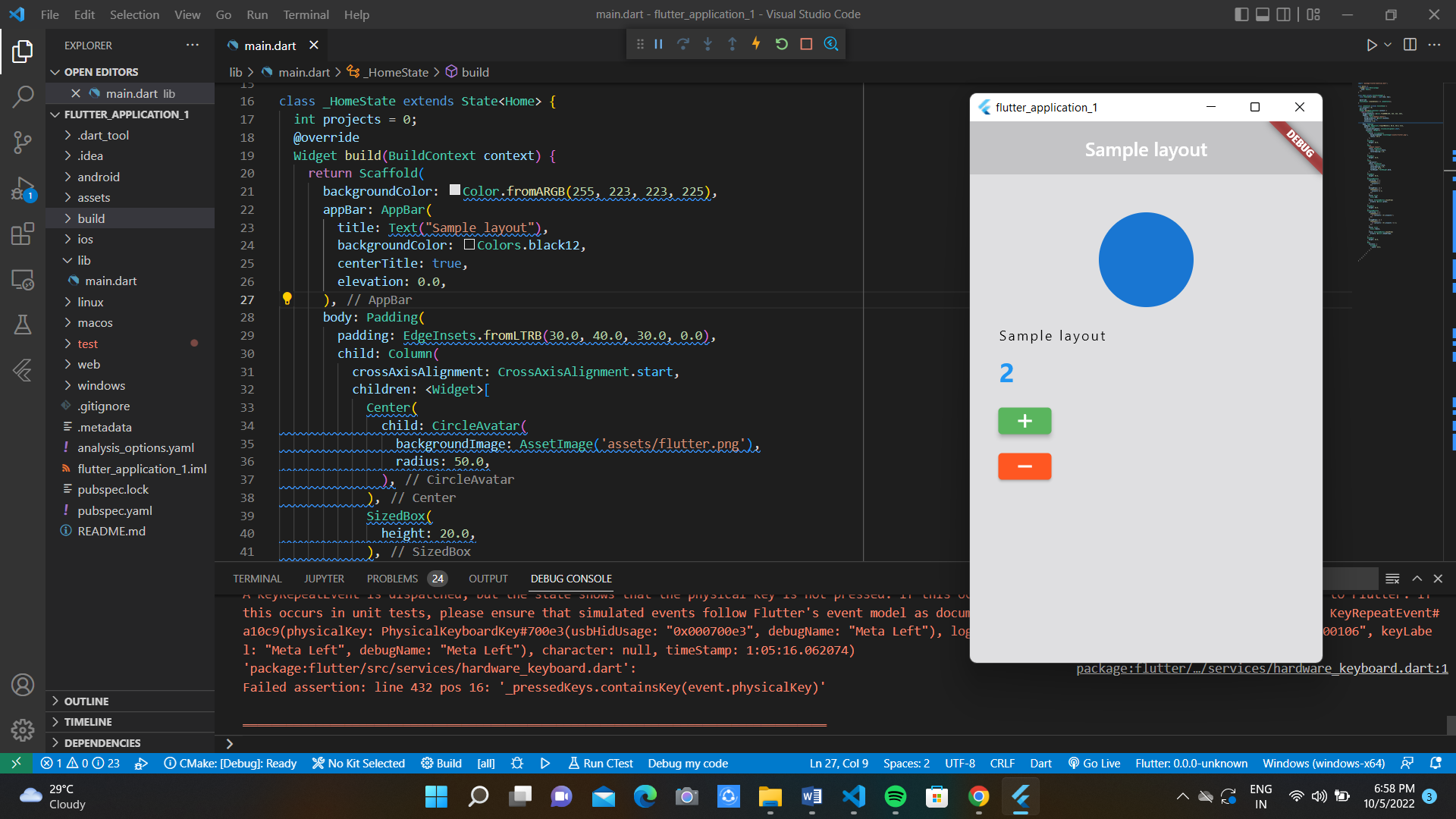
),

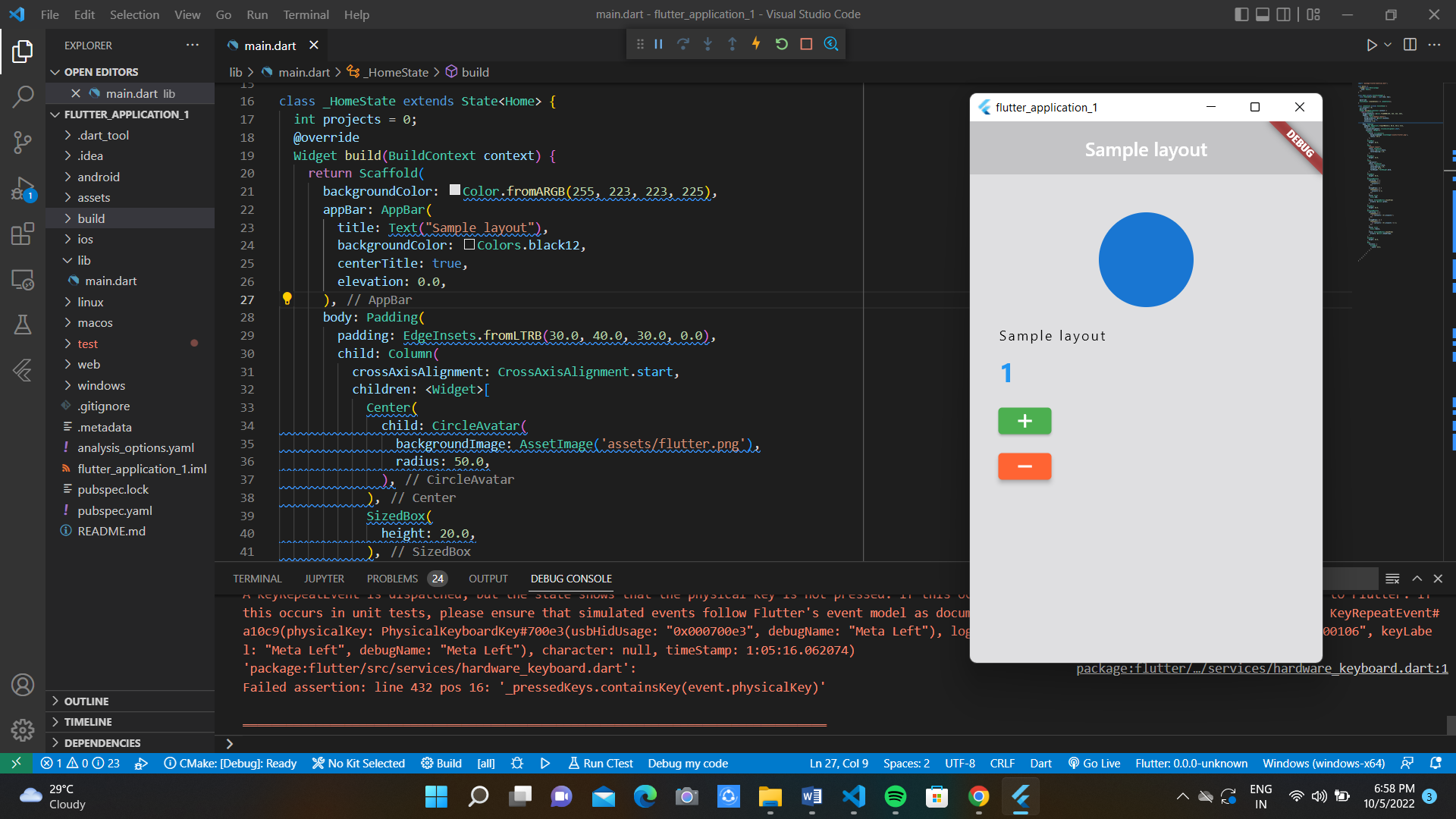
);

}

}

**Output:**

**** ****



**Result:**

An application that uses layout managers and event listeners has been implemented successfully.

**Ex: 3 Creation of Calculator Application**

**Date: 01/09/2022**

**Aim:**

To create a mobile calculator application

**Code:**

import 'package:flutter/material.dart';

void main() => runApp(const MyApp());

class MyApp extends StatelessWidget {

const MyApp({Key? key}) : super(key: key);

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'Calculator',

theme: ThemeData(

primarySwatch: Colors.blue,

),

debugShowCheckedModeBanner: false,

home: const MyHomePage(),

);

}

}

class MyHomePage extends StatefulWidget {

const MyHomePage({Key? key}) : super(key: key);

@override

\_MyHomePageState createState() => \_MyHomePageState();

}

class \_MyHomePageState extends State<MyHomePage> {

String output = "0";

String \_output = "0";

double num1 = 0.0;

double num2 = 0.0;

String operand = "";

buttonPressed(String buttonText) {

if (buttonText == "CLEAR") {

\_output = "0";

num1 = 0.0;

num2 = 0.0;

operand = "";

} else if (buttonText == "+" ||

buttonText == "-" ||

buttonText == "/" ||

buttonText == "X") {

num1 = double.parse(output);

operand = buttonText;

\_output = "0";

} else if (buttonText == ".") {

if (\_output.contains(".")) {

return;

} else {

\_output = \_output + buttonText;

}

} else if (buttonText == "=") {

num2 = double.parse(output);

if (operand == "+") {

\_output = (num1 + num2).toString();

}

if (operand == "-") {

\_output = (num1 - num2).toString();

}

if (operand == "X") {

\_output = (num1 \* num2).toString();

}

if (operand == "/") {

\_output = (num1 / num2).toString();

}

num1 = 0.0;

num2 = 0.0;

operand = "";

} else {

\_output = \_output + buttonText;

}

setState(() {

output = double.parse(\_output).toStringAsFixed(2);

});

}

Widget buildButton(String buttonText) {

return Expanded(

child: OutlinedButton(

style: OutlinedButton.styleFrom(

shape: RoundedRectangleBorder(

borderRadius: BorderRadius.circular(0.0),

),

side: const BorderSide(width: 1, color: Colors.grey),

minimumSize: const Size.fromHeight(50.0), // Set this

padding: EdgeInsets.zero, // and this

),

child: Text(

buttonText,

style: const TextStyle(fontSize: 20.0, fontWeight: FontWeight.bold),

),

onPressed: () => buttonPressed(buttonText),

));

}

@override

Widget build(BuildContext context) {

return Scaffold(

appBar: AppBar(

title: const Text("Calculator"),

),

body: Column(

children: <Widget>[

const Expanded(

child: Divider(

color: Colors.white,

),

),

Column(children: [

Container(

alignment: Alignment.centerRight,

padding: const EdgeInsets.symmetric(

vertical: 24.0, horizontal: 12.0),

child: Text(output,

style: const TextStyle(

fontSize: 48.0,

fontWeight: FontWeight.bold,

))),

Row(children: [

buildButton("7"),

buildButton("8"),

buildButton("9"),

buildButton("/")

]),

Row(children: [

buildButton("4"),

buildButton("5"),

buildButton("6"),

buildButton("X")

]),

Row(children: [

buildButton("1"),

buildButton("2"),

buildButton("3"),

buildButton("-")

]),

Row(children: [

buildButton("."),

buildButton("0"),

buildButton("00"),

buildButton("+")

]),

Row(children: [

buildButton("CLEAR"),

buildButton("="),

])

])

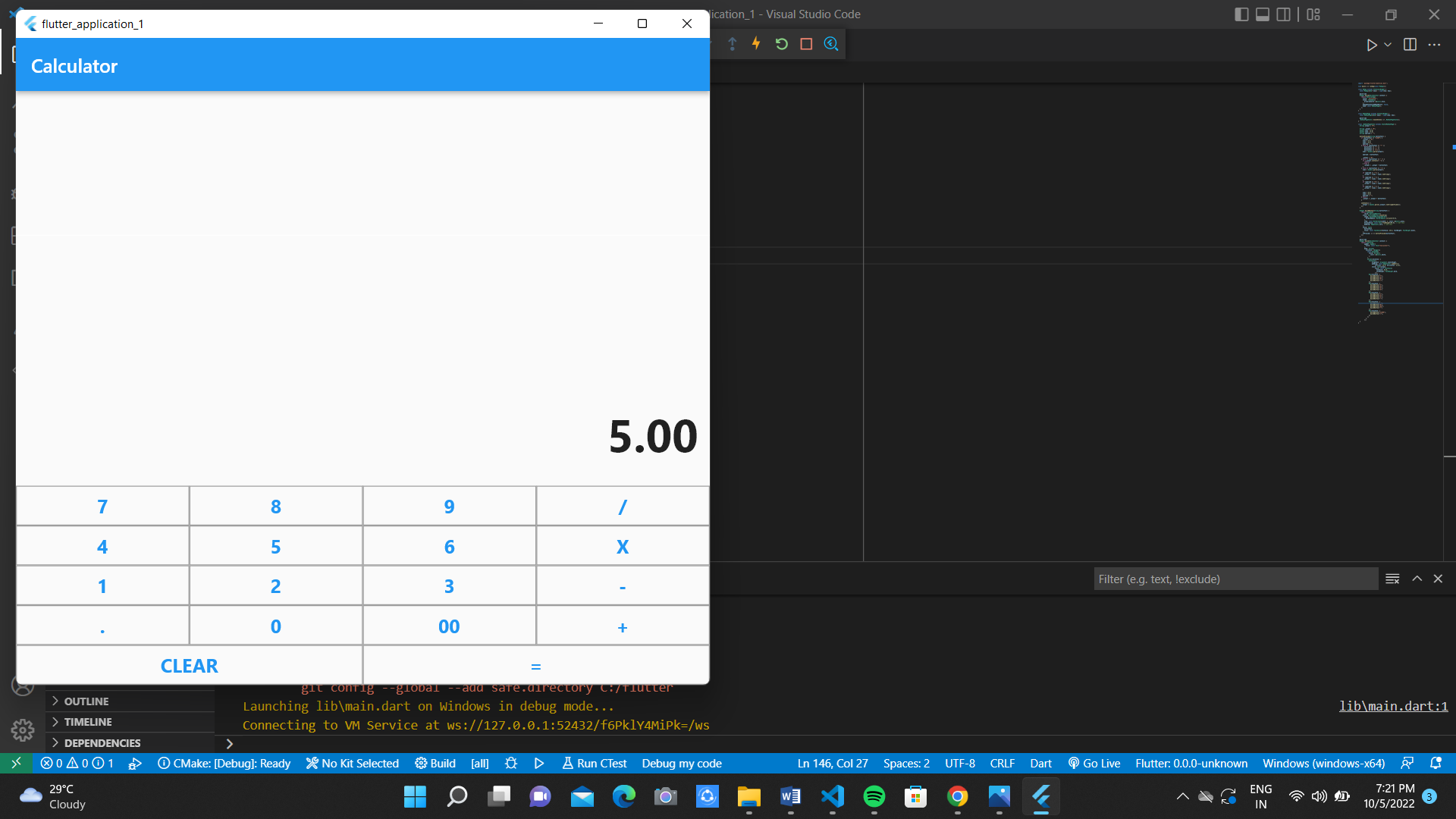
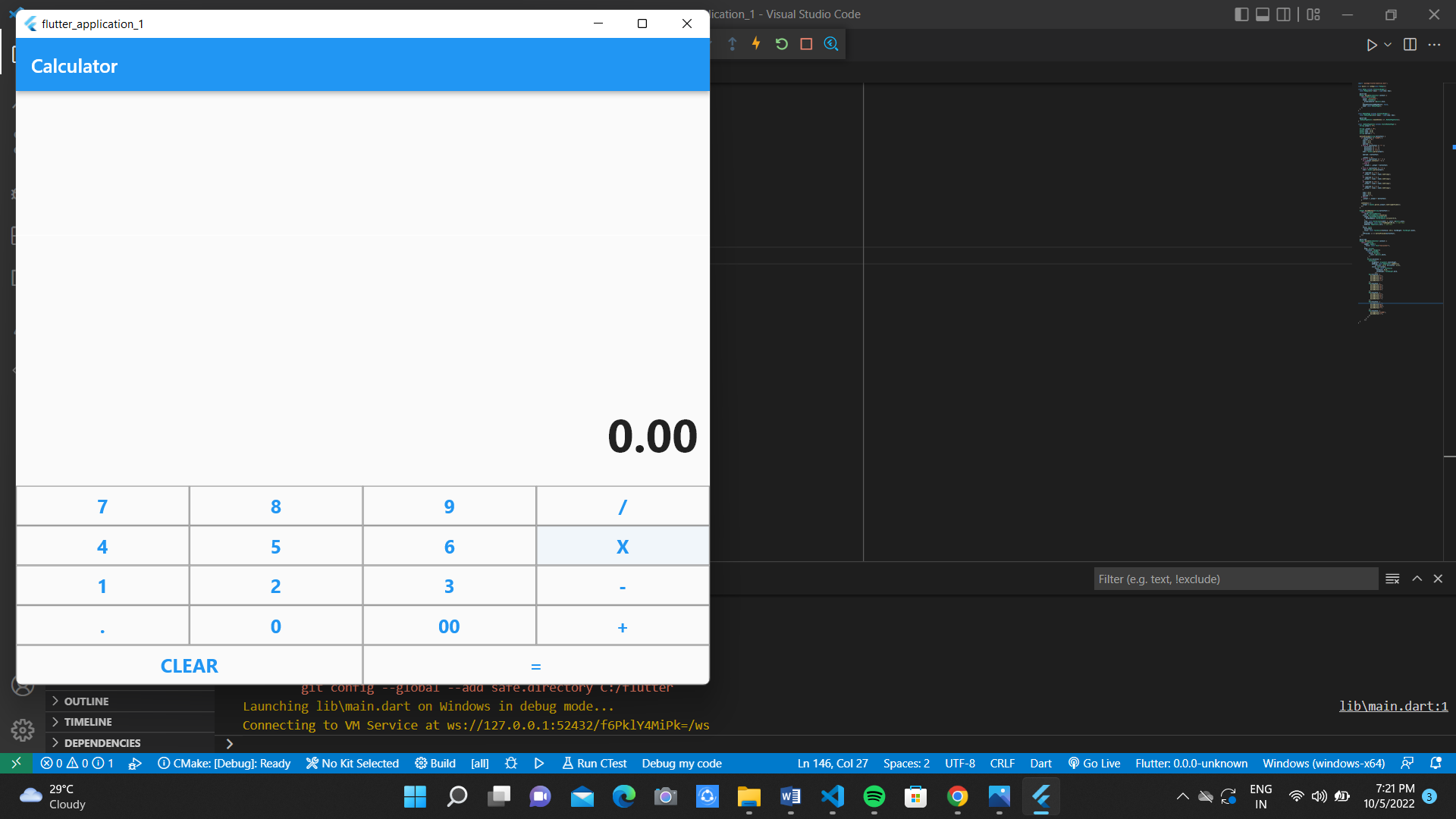
],

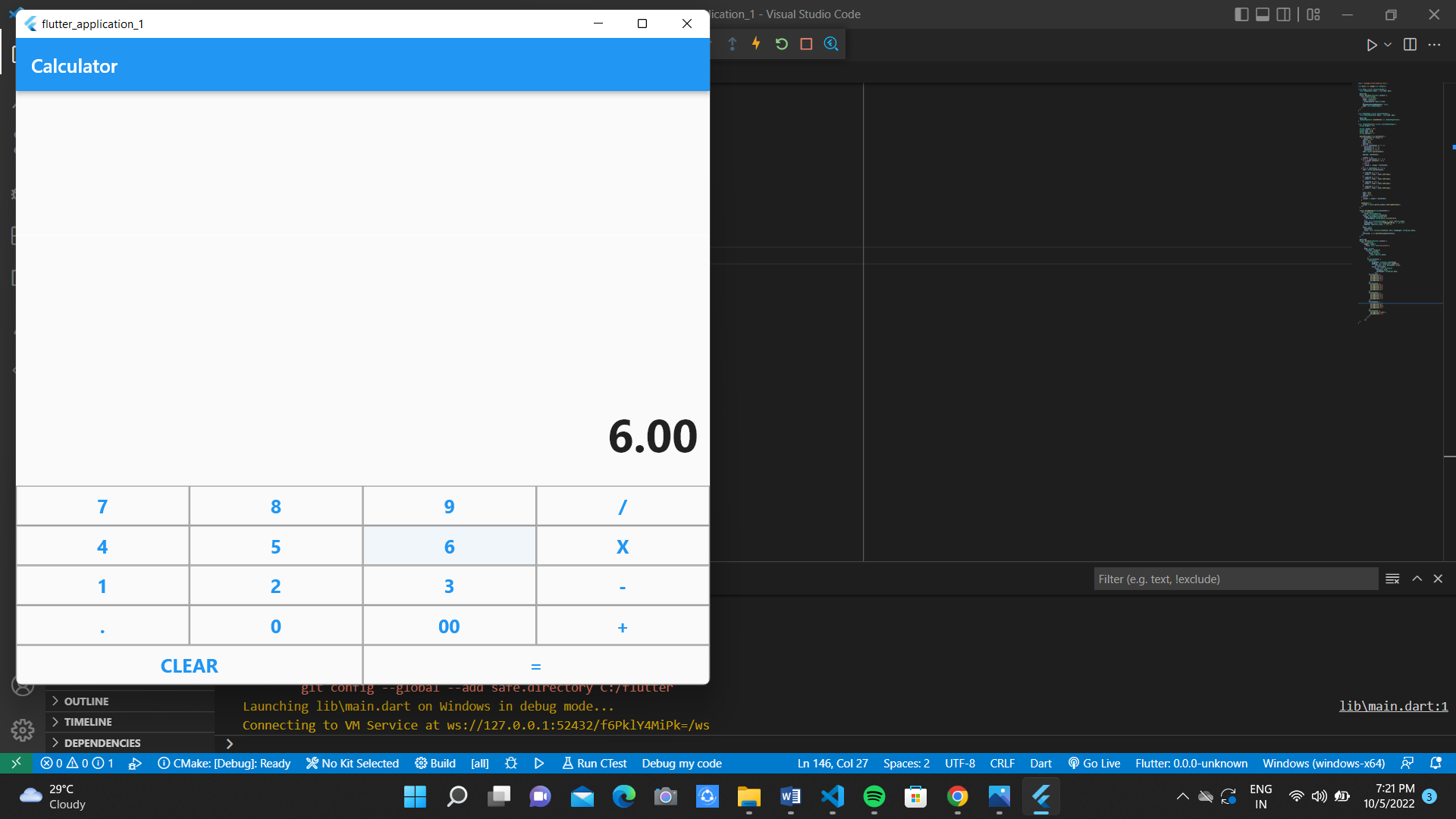
));

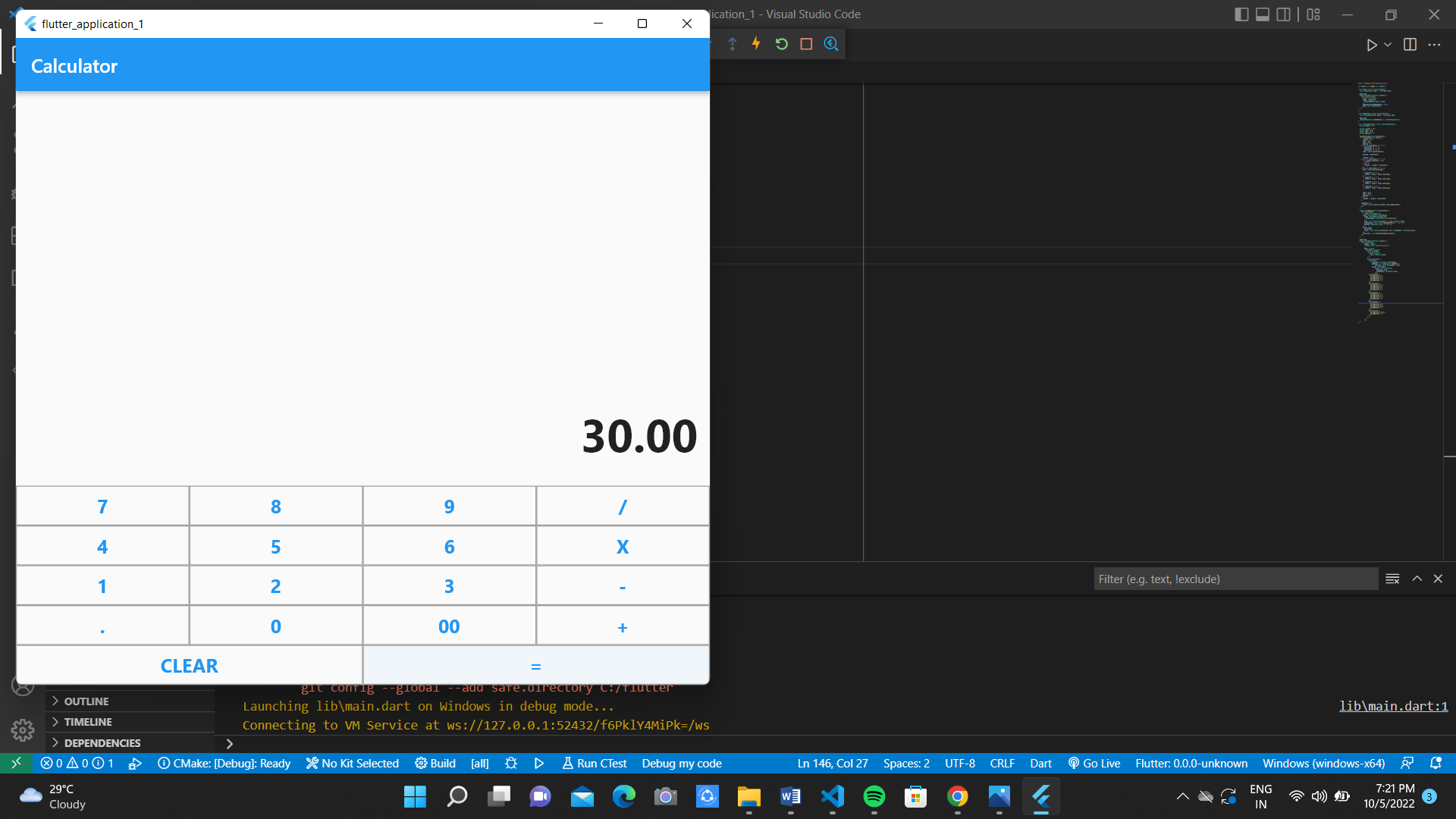
}

}

**Output:**

**** ****

****

****

**Result:**

A calculator application for mobiles has been implemented successfully.

**Ex: 4 An application that draws basic graphical primitives on screen**

**Date: 08/09/2022**

**Aim:**

To create a mobile application that draws basic graphical primitives on screen.

**Code:**

import 'package:flutter/material.dart';

final Color darkBlue = Color.fromARGB(255, 18, 32, 47);

void main() {

runApp(MyApp());

}

class MyApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

theme: ThemeData.dark().copyWith(scaffoldBackgroundColor: darkBlue),

debugShowCheckedModeBanner: false,

home: Scaffold(

// Outer white container with padding

body: Container(

color: Colors.black,

padding: EdgeInsets.symmetric(horizontal: 40, vertical: 80),

// Inner yellow container

child: Container(

// pass double.infinity to prevent shrinking of the painter area to 0.

width: double.infinity,

height: double.infinity,

color: Color.fromARGB(255, 126, 125, 125),

child: CustomPaint(painter: FaceOutlinePainter()),

),

),

),

);

}

}

class FaceOutlinePainter extends CustomPainter {

@override

void paint(Canvas canvas, Size size) {

final paint = Paint();

paint.style = PaintingStyle.stroke;

paint.strokeWidth = 4.0;

paint.color = Color.fromARGB(255, 244, 67, 54);

canvas.drawOval(

Rect.fromLTWH(size.width - 120, 40, 100, 100),

paint,

);

canvas.drawRect(

Rect.fromLTWH(20, 40, 100, 100),

paint,

);

final mouth = Path();

mouth.moveTo(size.width \* 0.8, size.height \* 0.6);

mouth.arcToPoint(

Offset(size.width \* 0.2, size.height \* 0.6),

radius: Radius.circular(150),

);

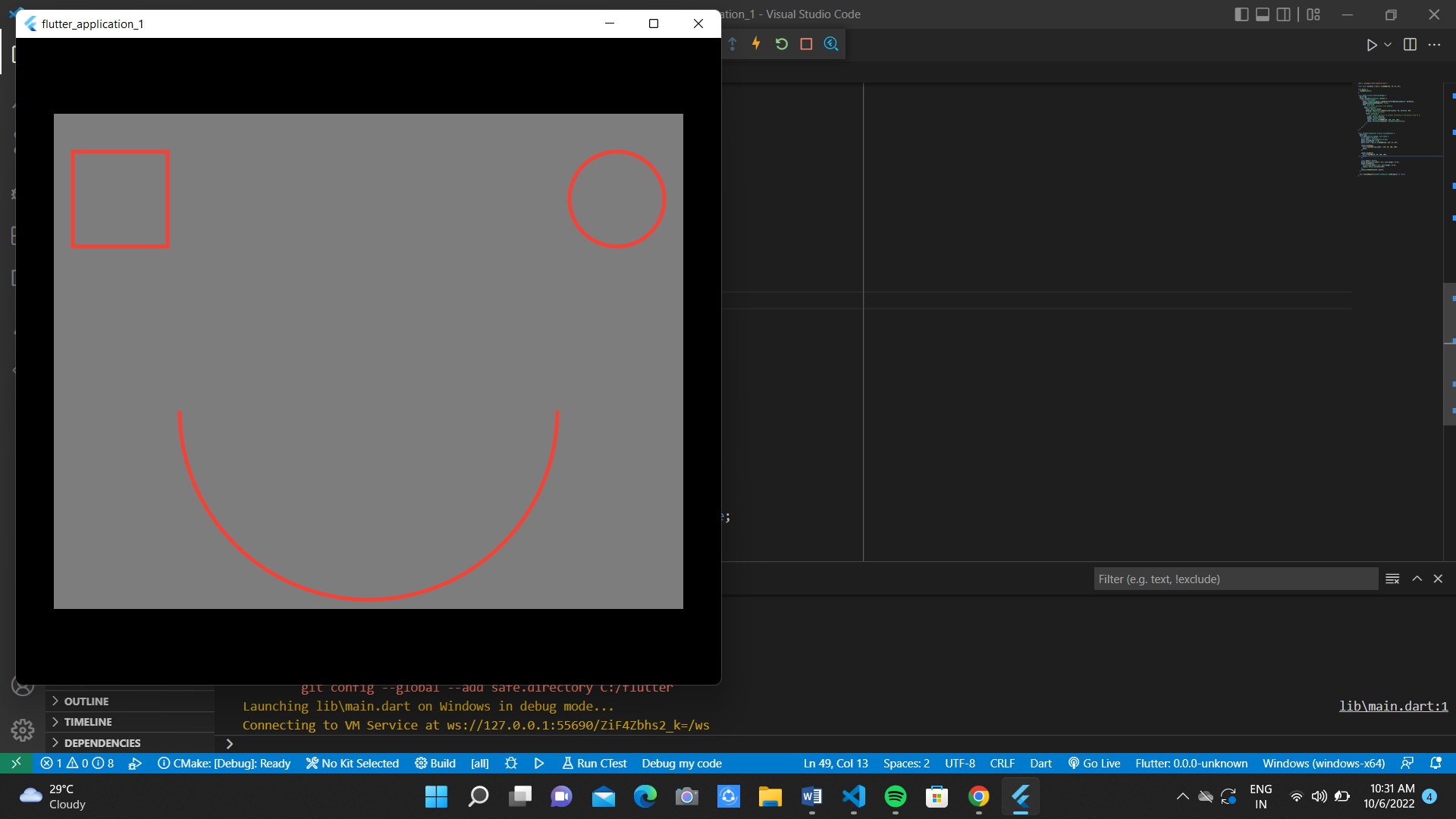
canvas.drawPath(mouth, paint);

}

bool shouldRepaint(FaceOutlinePainter oldDelegate) => false;

}

**Output:**

****

**Result:**

A mobile application that draws basic graphical primitives on screen has been implemented successfully.