import 'package:flutter/material.dart';

void main() {

runApp(CGPACalculatorApp());

}

class CGPACalculatorApp extends StatelessWidget {

@override

Widget build(BuildContext context) {

return MaterialApp(

title: 'CGPA Calculator',

theme: ThemeData(

primarySwatch: Colors.blue,

),

home: CGPACalculatorScreen(),

);

}

}

class CGPACalculatorScreen extends StatefulWidget {

@override

\_CGPACalculatorScreenState createState() => \_CGPACalculatorScreenState();

}

class \_CGPACalculatorScreenState extends State<CGPACalculatorScreen> {

final List<TextEditingController> creditControllers = [];

final List<TextEditingController> markControllers = [];

double cgpa = 0.0;

int numberOfSubjects = 1;

// Function to calculate CGPA

void calculateCGPA() {

double totalCredits = 0.0;

double totalPoints = 0.0;

for (int i = 0; i < numberOfSubjects; i++) {

double credits = double.tryParse(creditControllers[i].text) ?? 0;

double marks = double.tryParse(markControllers[i].text) ?? 0;

double points = getGradePointsFromMarks(marks);

totalCredits += credits;

totalPoints += points \* credits;

}

setState(() {

cgpa = totalCredits != 0 ? totalPoints / totalCredits : 0.0;

});

}

// Function to map marks to grade points

double getGradePointsFromMarks(double marks) {

if (marks >= 90) {

return 4.0; // A

} else if (marks >= 80) {

return 3.7; // A-

} else if (marks >= 70) {

return 3.3; // B+

} else if (marks >= 60) {

return 3.0; // B

} else if (marks >= 50) {

return 2.7; // B-

} else if (marks >= 40) {

return 2.0; // C

} else if (marks >= 33) {

return 1.0; // D

} else {

return 0.0; // F

}

}

@override

Widget build(BuildContext context) {

return Scaffold(

backgroundColor: Colors.blueGrey, // Sky blue background

body: Center(

child: Container(

width: 650, // Fixed width for square shape

height: 550, // Fixed height for square shape

decoration: BoxDecoration(

color: Colors.white, // White container background

borderRadius: BorderRadius.zero, // Square edges

boxShadow: [

BoxShadow(

color: Colors.black26,

blurRadius: 10,

offset: Offset(0, 5),

),

],

),

padding: const EdgeInsets.all(16.0),

child: Column(

mainAxisAlignment: MainAxisAlignment.center,

children: [

Text(

'Enter Your Subject Details',

style: TextStyle(

fontSize: 18,

fontWeight: FontWeight.bold,

color: Colors.black, // Black text color

),

textAlign: TextAlign.center,

),

Expanded(

child: ListView.builder(

shrinkWrap: true,

itemCount: numberOfSubjects,

itemBuilder: (context, index) {

if (creditControllers.length <= index) {

creditControllers.add(TextEditingController());

markControllers.add(TextEditingController());

}

return Padding(

padding: const EdgeInsets.symmetric(vertical: 4.0),

child: Row(

children: [

Expanded(

child: TextField(

controller: creditControllers[index],

keyboardType: TextInputType.number,

decoration: InputDecoration(

labelText: 'Credits',

enabledBorder: OutlineInputBorder(

borderRadius: BorderRadius.zero, // Square edges

borderSide: BorderSide(color: Colors.grey),

),

focusedBorder: OutlineInputBorder(

borderRadius: BorderRadius.zero, // Square edges

borderSide: BorderSide(color: Colors.blue),

),

),

),

),

SizedBox(width: 10),

Expanded(

child: TextField(

controller: markControllers[index],

keyboardType: TextInputType.number,

decoration: InputDecoration(

labelText: 'Marks',

enabledBorder: OutlineInputBorder(

borderRadius: BorderRadius.zero, // Square edges

borderSide: BorderSide(color: Colors.grey),

),

focusedBorder: OutlineInputBorder(

borderRadius: BorderRadius.zero, // Square edges

borderSide: BorderSide(color: Colors.blue),

),

),

),

),

],

),

);

},

),

),

SizedBox(height: 10),

ElevatedButton(

onPressed: calculateCGPA,

style: ElevatedButton.styleFrom(

shape: RoundedRectangleBorder(

borderRadius: BorderRadius.zero, // Square button

),

),

child: Text('Calculate CGPA'),

),

SizedBox(height: 10),

Text(

'CGPA: ${cgpa.toStringAsFixed(2)}',

style: TextStyle(

fontSize: 16,

fontWeight: FontWeight.bold,

color: Colors.black, // Black text color for CGPA result

),

),

SizedBox(height: 10),

ElevatedButton(

onPressed: () {

setState(() {

numberOfSubjects++;

});

},

style: ElevatedButton.styleFrom(

shape: RoundedRectangleBorder(

borderRadius: BorderRadius.zero, // Square button

),

),

child: Text('Add More Subjects'),

),

],

),

),

),

);

}

}