EX-2 Flutter application with Widgets, GUI Components, Fonts, and Colors

Aim

To develop a flutter application that uses Widgets, GUI Components, Fonts and Colors.

Definitions

Flutter

Flutter is not a programming language. It's a software development kit (SDK) with prewritten code, consisting of ready-to-use and customizable widgets, as well as libraries, tools, and documentation that together serve to build cross-platform apps.

Flutter plugin

A Flutter plugin is a special kind of package that enables Flutter apps to interact with platform-specific APIs (iOS, Android, web, desktop). Plugins can include Dart code, but crucially, they also contain platform-specific implementation code written in Kotlin/Java for Android and Swift/Obj-C for iOS.

Dart plugin

The Dart plugin adds Dart support to IntelliJ Platform-based IDEs developed by JetBrains. These IDEs provide features unique to specific development technologies. The IDEs recommended for Dart and Flutter development include: IntelliJ IDEA which specializes in JVM-based language development.

Flutter SDK

Flutter is Google's free, open-source software development kit (SDK) for cross-platform mobile application development. Using a single platform-agnostic codebase, Flutter helps developers build high-performance, scalable applications with attractive and functional user interfaces for Android or IOS.

Widgets

Widgets are an essential aspect of home screen customization. You can think of them as "at-a-glance" views of an app's most important data and functionality that are accessible right on the user's home screen.

GUI Components

The main pieces of a GUI are a pointer, icons, windows, menus, scroll bars, and an intuitive input device. Some common GUIs are the ones associated with Microsoft Windows, Mac OSX, Chrome OS, GNOME, KDE, and Android.

Procedure

1. **Open android studio**
2. **Click ‘new flutter project’**
3. **Select ‘flutter’ at the left side of the window**
4. **Add ‘flutter sdk’ from the desired location**
5. **Click ‘next’ and specify the project name and select language ‘java’, check only Android, Web and Windows under platforms then click ‘create’**
6. **Create a new dart file under ‘lib’ folder in the projects window (right click over lib folder - > new -> dart file -> specify the file name as ‘counter’ -> press ‘enter’**
7. **Type the following codes in the counter.dart file**

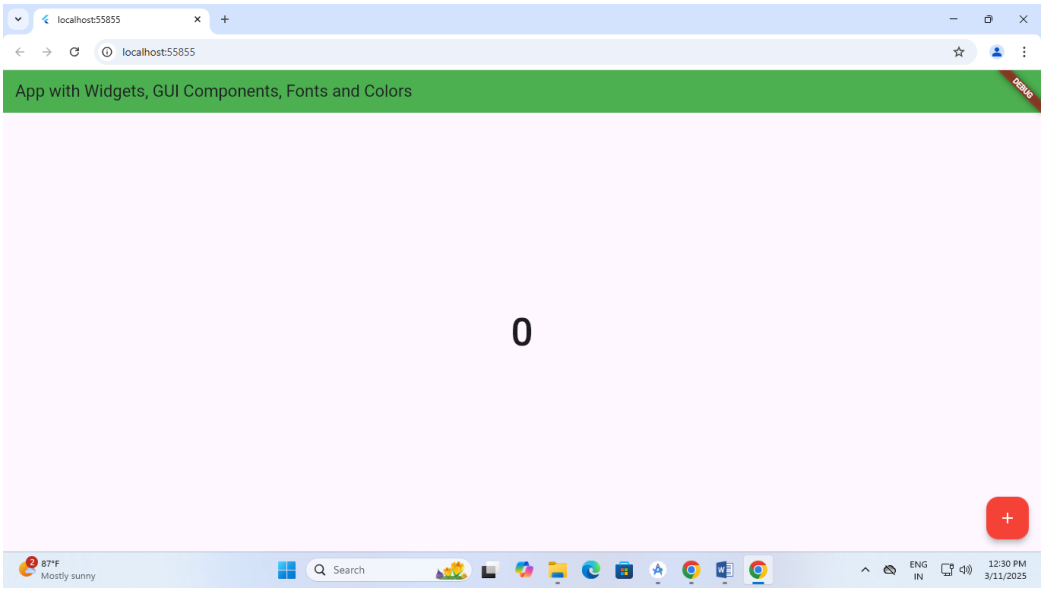
Counter.dart

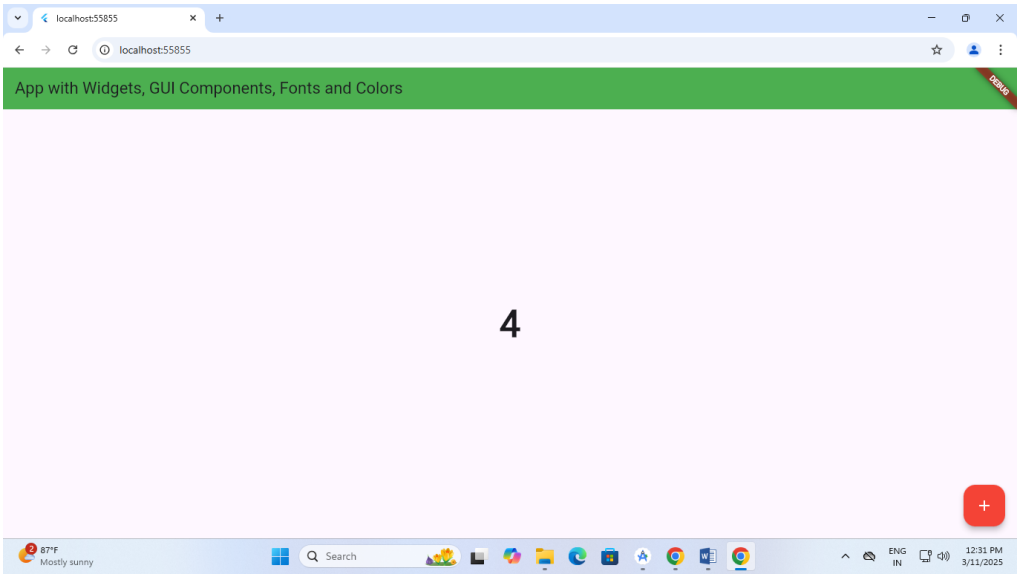
```
import 'package:flutter/material.dart';
void main() => runApp(const Counter());

// MyApp is the root widget of the application
class Counter extends StatelessWidget {
  const Counter({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      home: HomePage(),
    );
  }
}

// HomePage is the main screen of the app
class HomePage extends StatefulWidget {
  const HomePage({Key? key}) : super(key: key);

  @override
  _HomePageState createState() => _HomePageState();
}

class _HomePageState extends State<HomePage> {
  int _counter = 0; // Variable to store the counter value
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        // Set the background color of the app bar
        backgroundColor: Colors.green,
        // Set the title of the app bar
        title: const Text("App with Widgets, GUI Components, Fonts and Colors"),
      ),
```

```
      // The main body of the scaffold
      body: Center(
        // Display a centered text widget
        child: Text(
          "$_counter",
          // Apply text styling
          style: TextStyle(
            fontSize: 50, // Set font size
            fontWeight: FontWeight.bold, // Set font weight
          ),
        ),
      ),
      floatingActionButton: FloatingActionButton(
        backgroundColor: Colors.red,
        onPressed: () {
          // Increment the counter value by 1 using setState
          setState(() {
            _counter++;
          });
        },
        child: Icon(
          Icons.add,
          color: Colors.white,
        ),
      ),
    );
  }
}
```

8. **Save the file counter.dart (click main menu -> saveall)**
9. **Select device as 'chrome(web)'**
10. **Click on run/debug configuration -> edit configurations ->specify dart file name (counter.dart) -> browse and set dart entrypoint as counter.dart -> click ok -> click run**

Output


localhost:55855
App with Widgets, GUI Components, Fonts and Colors
DEBUG
0
+
87°F
Mostly sunny
Search
ENG
IN
12:30 PM
3/11/2025


localhost:55855
App with Widgets, GUI Components, Fonts and Colors
DEBUG
4
+
87°F
Mostly sunny
Search
ENG
IN
12:31 PM
3/11/2025

Result

Thus, a flutter application that uses Widgets, GUI Components, Fonts and Colors has been developed.