

GUIA FLUTTER #1

PAULA ANDREA CONDE AGUJA

FICHA. 3197815

Diego Alejandro Boada Morales

Aplicaciones Móviles II

SEDE SENA

CENTRO DE SERVICIOS FINANCIEROS

BOGOTA D.C.

6 DE DICIEMBRE 2025

1. Abre el archivo lib/main.dart y reemplázalo con el siguiente código:

The screenshot shows the Android Studio interface. On the left, the project structure is visible with the main.dart file selected. The code editor contains the following Dart code:

```
1 import 'package:flutter/material.dart';
2
3 void main() => runApp(MyApp());
4
5 class MyApp extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8     return MaterialApp(
9       home: Scaffold(
10         appBar: AppBar(
11           title: Text("Flutter Scaffold"),
12         ), // AppBar
13         body: Center(
14           child: Text("Holaaa FLuterrrr!!!"),
15         ), // Center
16       ), // Scaffold
17     ); // MaterialApp
18   }
19 }
```

To the right, a browser window titled "localhost:55122" displays the text "Holaaa FLuterrrr!!!".

2. En el archivo lib/main.dart, cambia el zódigo dentro de body a lo siguiente

The screenshot shows the Android Studio interface again. The main.dart file has been modified. The code editor now contains:

```
1 import 'package:flutter/material.dart';
2
3 void main() => runApp(MyApp());
4
5 class MyApp extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8     return MaterialApp(
9       home: Scaffold(
10         appBar: AppBar(
11           title: Text("Flutter Scaffold"),
12         ), // AppBar
13         body: Center(
14           child: Container(
15             color: Colors.blue,
16             height: 150.0,
17             width: 150.0,
18             child: Center(child: Text("Holaaaaaa, how are you?")),
19           ), // Container
20         ), // Center
21       ), // Scaffold
22     ); // MaterialApp
23   }
24 }
```

The browser window now displays a blue square containing the text "Holaaaaaa, how are you?".

3. El código anterior ya utiliza Center y Text. Modifica el código para personalizar el texto y su estilo

The screenshot shows a Flutter application running in a browser window titled "Flutter Scaffold". The page content is "Flutter Scaffold" with a single line of blue text: "Taller flutter". The browser address bar shows "localhost:55785". The code editor on the left shows the main.dart file with the following code:

```
1 import 'package:flutter/material.dart';
2
3 void main() => runApp(MyApp());
4
5 class MyApp extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8     return MaterialApp(
9       home: Scaffold(
10         appBar: AppBar(
11           title: Text('Flutter Scaffold'),
12         ),
13         body: Center(
14           child: Text('Taller flutter',
15             style: TextStyle(fontSize: 30, color: Colors.blue),
16         ),
17       ),
18     );
19   }
20 }
21
22 
```

The terminal at the bottom of the screen shows the command "flutter run" being executed.

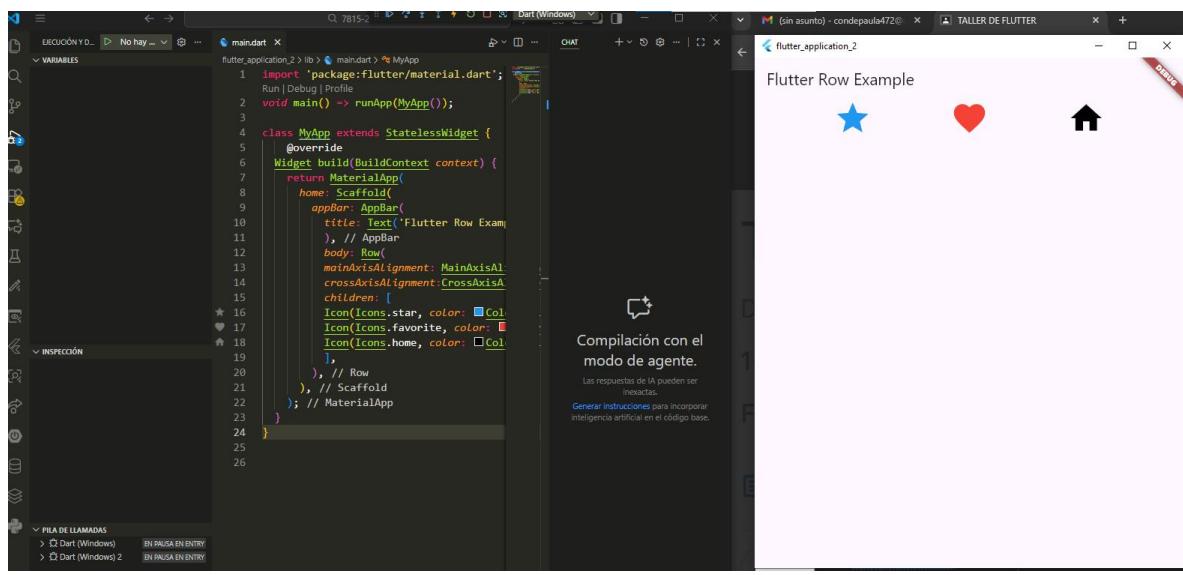
4. Modifica el body para utilizar una columna que contenga varios textos

The screenshot shows a Flutter application running in a browser window titled "Flutter Scaffold". The page content consists of three lines of text: "Cold, Cold Feeling" in black, "T-Bone Shuffle" in blue, and "I Ain t Drunk" in red. The browser address bar shows "localhost:56134". The code editor on the left shows the main.dart file with the following code:

```
1 import 'package:flutter/material.dart';
2
3 void main() => runApp(MyApp());
4
5 class MyApp extends StatelessWidget {
6   @override
7   Widget build(BuildContext context) {
8     return MaterialApp(
9       home: Scaffold(
10         appBar: AppBar(
11           title: Text('Flutter Scaffold'),
12         ),
13         body: Column(
14           mainAxisAlignment: MainAxisAlignment.center,
15           crossAxisAlignment: CrossAxisAlignment.center,
16           children: [
17             Text(
18               "Cold, Cold Feeling",
19               style: TextStyle(fontSize: 24, color: Colors.black),
20             ),
21             Text(
22               "T-Bone Shuffle",
23               style: TextStyle(fontSize: 24, color: Colors.blue),
24             ),
25             Text(
26               "I Ain t Drunk",
27               style: TextStyle(fontSize: 24, color: Colors.red),
28             ),
29           ],
30         ),
31       );
32     }
33   }
34 }
```

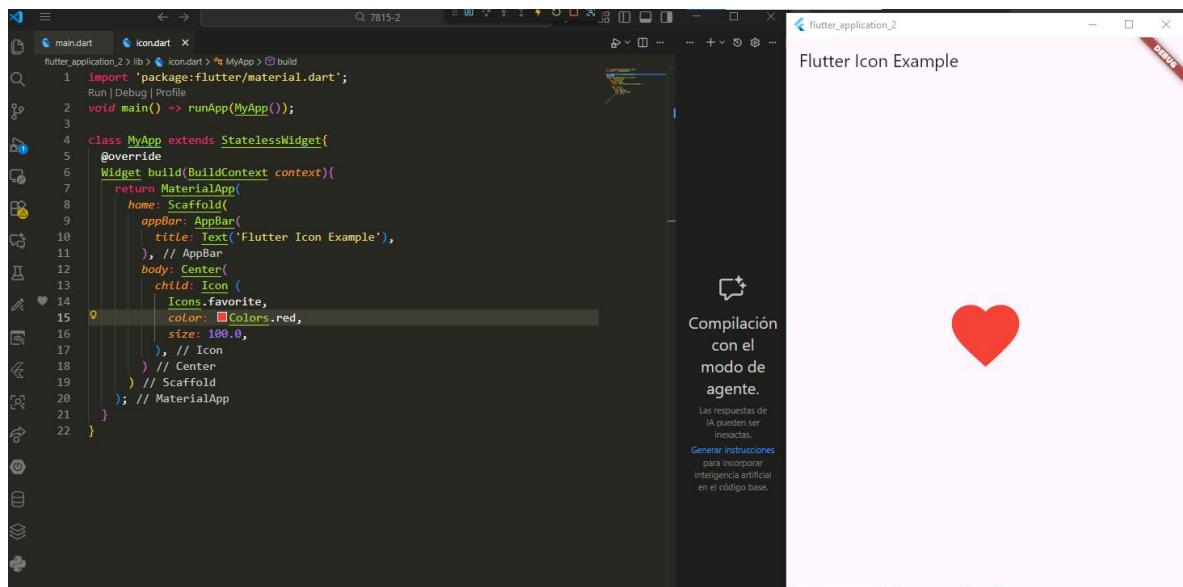
The terminal at the bottom of the screen shows the command "flutter run" being executed.

5. Row: Similar a Column, pero organiza los widgets en una fila horizontal



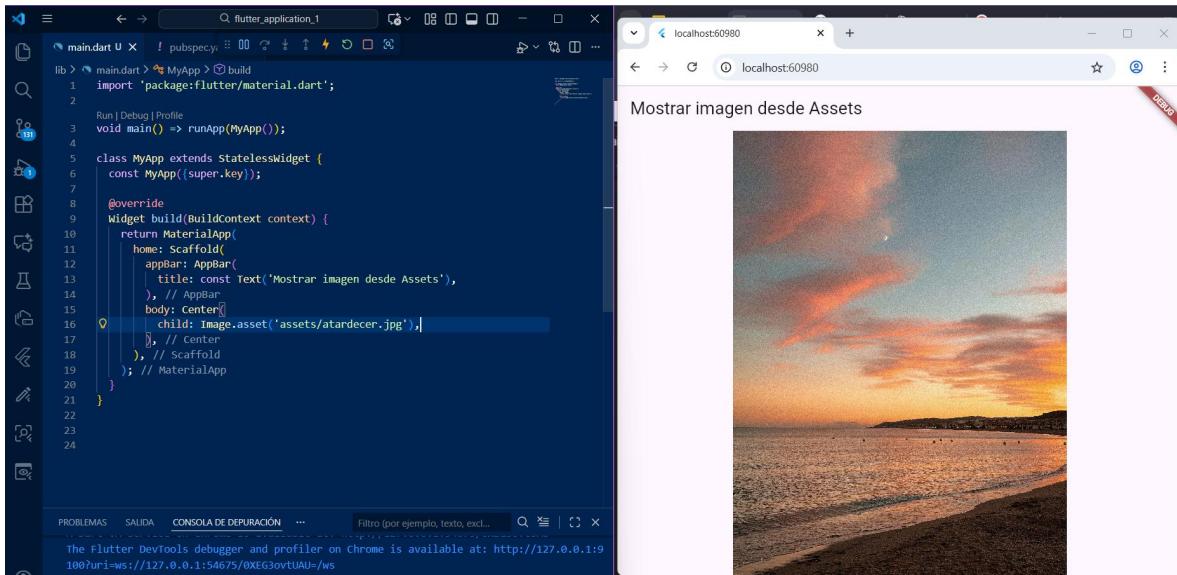
The screenshot shows the VS Code interface with two panes. The left pane displays the code for `main.dart`:import 'package:flutter/material.dart';void main() => runApp(MyApp());class MyApp extends StatelessWidget{@overrideWidget build(BuildContext context){return MaterialApp(home: Scaffold(appBar: AppBar(title: Text('Flutter Row Example')), body: Row(mainAxisAlignment: MainAxisAlignment.spaceEvenly, crossAxisAlignment: CrossAxisAlignment.baseline, mainAxisAlignment: MainAxisAlignment.end, children: [Icon(Icons.star, color: Colors.blue), Icon(Icons.favorite, color: Colors.red), Icon(Icons.home, color: Colors.black)],),),);}}The right pane shows the running application titled "Flutter Row Example" with three icons: a blue star, a red heart, and a black house.

6. Icon: Permite agregar íconos a tu interfaz. Flutter tiene un set predefinido de íconos disponibles en la clase Icons.

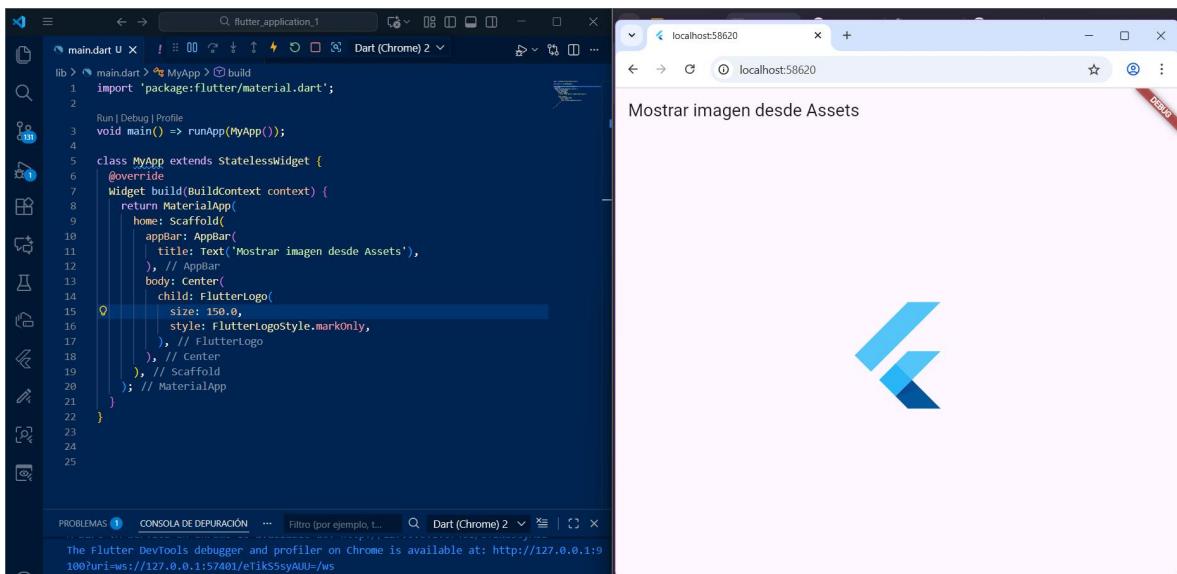


The screenshot shows the VS Code interface with two panes. The left pane displays the code for `main.dart` and `icon.dart`:import 'package:flutter/material.dart';void main() => runApp(MyApp());class MyApp extends StatelessWidget{@overrideWidget build(BuildContext context){return MaterialApp(home: Scaffold(appBar: AppBar(title: Text('Flutter Icon Example')), body: Center(child: Icon(Icons.favorite, color: Colors.red, size: 100.0,),),),);}}The right pane shows the running application titled "Flutter Icon Example" with a large red heart icon.

7. Una vez que hecho los pasos anteriores, pueden utilizar la imagen en el código de la siguiente manera:



8. **FlutterLogo**: Es un widget que muestra el logo de Flutter. Puede ser útil para marcas o temas relacionados con Flutter.



9. ListView y RaisedButton:

ListView: Un widget que permite mostrar una lista de widgets desplazables vertical u horizontalmente.

RaisedButton: Un botón elevado (ha sido reemplazado por ElevatedButton en versiones más recientes de Flutter).

The screenshot shows the Android Studio interface with two main panes. The left pane displays the Dart code for a Flutter application. The right pane shows the browser output at localhost:54099, which displays a list view containing four elevated buttons labeled Botón 1 through Botón 4. Each button has an associated print statement in its onPressed handler. The bottom of the screen shows the terminal output, which shows the application was restarted in 3.249ms and each button's press was printed.

```
1 import 'package:flutter/material.dart';
2 Run | Debug | Profile
3 void main() => runApp(MyApp());
4 class MyApp extends StatelessWidget {
5   @override
6   Widget build(BuildContext context) {
7     return MaterialApp(
8       home: Scaffold(
9         appBar: AppBar(
10           title: Text('ListView y ElevatedButton'),
11         ), // AppBar
12         body: ListView(
13           children: [
14             ElevatedButton(onPressed: (){
15               print("Botón 1 presionado");
16             },),
17             child: Text('Botón 1 '),
18             ElevatedButton(onPressed: (){
19               print("Botón 2 presionado");
20             },),
21             child: Text('Botón 2 '),
22             ElevatedButton(onPressed: (){
23               print("Botón 3 presionado");
24             },),
25             child: Text('Botón 3 '),
26             ElevatedButton(onPressed: (){
27               print("Botón 4 presionado");
28             },),
29             child: Text('Botón 4 '),
30           ],
31         ), // ListView
32       ), // Scaffold
33     ); // MaterialApp
34   }
35 }
36 
```

PROBLEMAS 5 SALIDA CONSOLA DE DEPURACIÓN TERMINAL PUERTOS

Restarted application in 3.249ms.

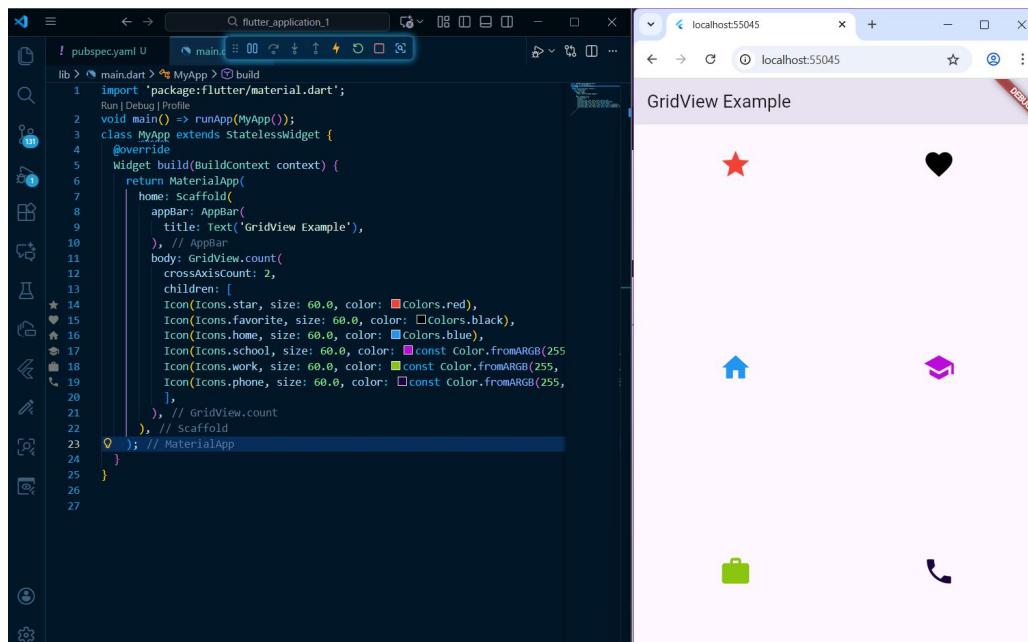
Botón 1 presionado

Botón 2 presionado

Botón 3 presionado

Botón 4 presionado

10. GridView: Similar a ListView, pero organiza los elementos en una cuadrícula, útil para crear layouts de múltiples columnas y filas.



11. Stack: Permite apilar widgets unos sobre otros, superponiéndolos, útil para crear interfaces más complejas como overlays.

The screenshot shows the VS Code interface with the code for a 'Stack Example'. The code uses a Stack widget to overlap two containers: a blue one at the top and a black one below it. A text widget is positioned within the black container. The browser preview on the right shows a blue background with a black rectangle on top containing the text 'Encima del contenedor'.

```
lib > main.dart > MyApp > build
1 import 'package:flutter/material.dart';
2 void main() => runApp(MyApp());
3 class MyApp extends StatelessWidget {
4   @override
5   Widget build(BuildContext context) {
6     return MaterialApp(
7       home: Scaffold(
8         appBar: AppBar(
9           title: Text('Stack Example'),
10        ), // AppBar
11        body: Stack(
12          children: [
13            Container(
14              color: Colors.blue,
15              height: 300,
16              width: 300,
17            ), // Container
18            Positioned(
19              top: 50,
20              left: 50,
21              child: Container(
22                color: Colors.black,
23                height: 100,
24                width: 100,
25              ), // Container
26            ), // Positioned
27            Positioned(
28              top: 100,
29              left: 100,
30              child: Text(
31                'Encima del contenedor',
32                style: TextStyle(color: Colors.amber, fontSize: 18),
33              ), // Text
34            ), // Positioned
35          ],
36        ), // Stack
37      ),
38    );
39  }
40}
```

12. Align: Alinea su hijo en una posición específica dentro de su contenedor padre, usando valores de alineación predefinidos o personalizados.

The screenshot shows the VS Code interface with the code for an 'Align Example'. It uses an Align widget to position a text child at the bottom right of its container. The browser preview on the right shows a light blue background with the text 'Alineado!' positioned in the bottom right corner.

```
lib > main.dart > ...
1 import 'package:flutter/material.dart';
2 void main() => runApp(MyApp());
3 class MyApp extends StatelessWidget {
4   @override
5   Widget build(BuildContext context) {
6     return MaterialApp(
7       home: Scaffold(
8         appBar: AppBar(
9           title: Text('Align Example'),
10        ), // AppBar
11        body: Container(
12          color: Colors.blue[50],
13          height: 300,
14          width: 300,
15          child: Align(
16            alignment: Alignment.bottomRight,
17            child: Text(
18              'Alineado!',
19              style: TextStyle(fontSize: 24, color: Colors.black),
20            ), // Text
21          ), // Align
22        ), // Container
23      ), // Scaffold
24    ); // MaterialApp
25  }
26}
27
```