



Desenvolvimento  
Mobile 1  
Aula 05

Prof. Me Daniel Vieira

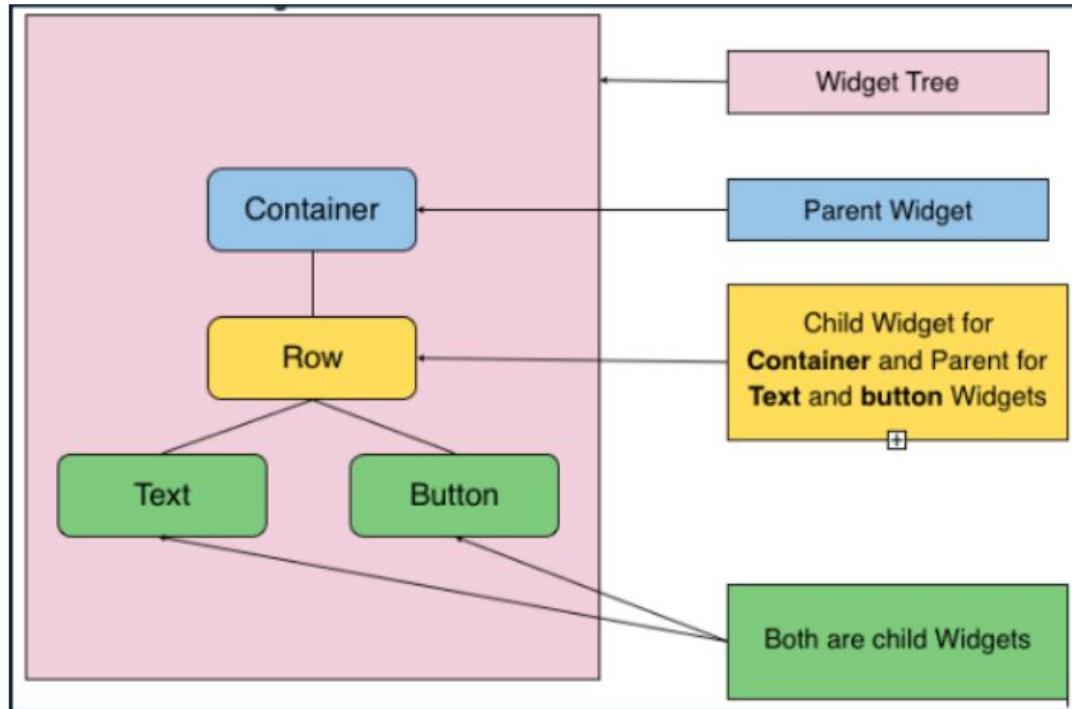
**SENAI**

# Agenda

- 1- Widgets
- 2-Criando projeto Flutter no VsCode
- 3 - Column e Rows
- 4 -Statefull e Stateless
- 5- Criando APP
- 6- Exercício

# Widgets

## São componentes do Flutter



# Widgets

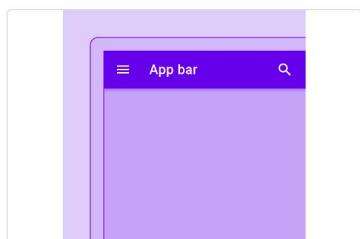
## São componentes do Flutter

### Basic widgets

UI > Widgets > Basics

Widgets you absolutely need to know before building your first Flutter app.

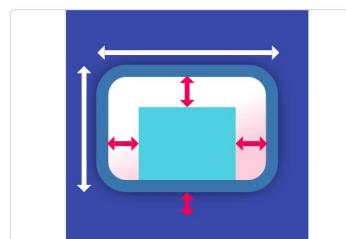
See more widgets in the [widget catalog](#).



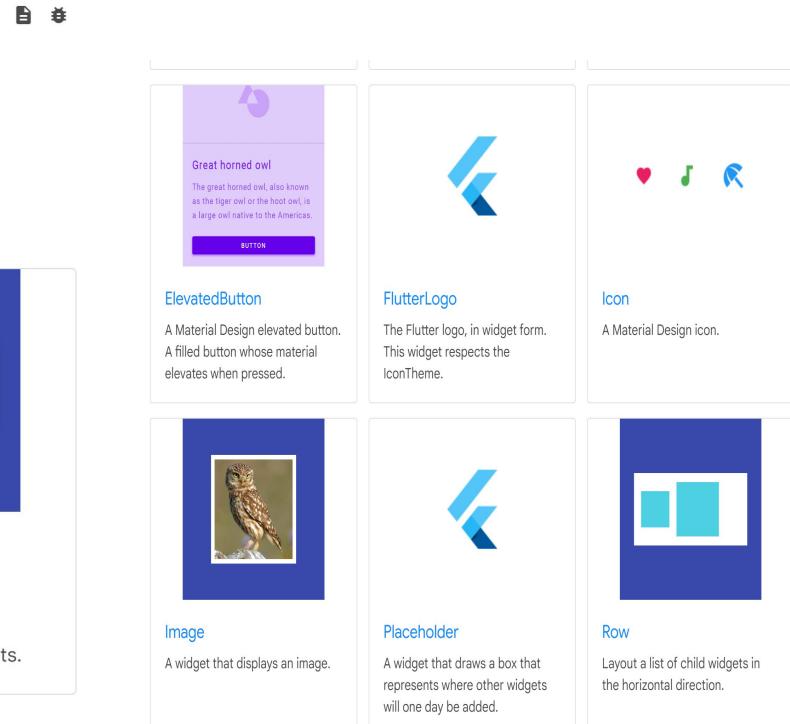
**AppBar**  
A toolbar that might contain other widgets such as a 'TabBar' and a 'FlexibleSpaceBar'.



**Column**  
Layout a list of child widgets in the vertical direction.



**Container**  
A convenience widget that combines common painting, positioning, and sizing widgets.

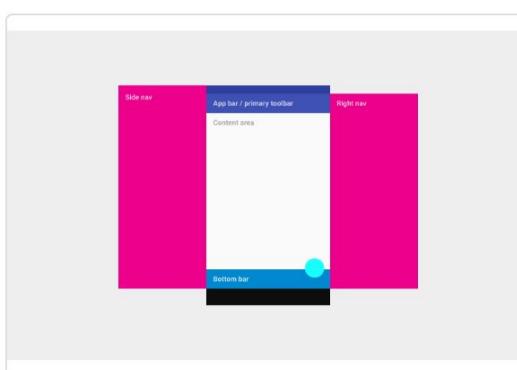


The screenshot shows a grid of six basic Flutter widgets:

- ElevatedButton**: A Material Design elevated button.
- FlutterLogo**: The Flutter logo, in widget form.
- Icon**: A Material Design icon.
- Image**: A widget that displays an image.
- Placeholder**: A widget that draws a box that represents where other widgets will one day be added.
- Row**: Layout a list of child widgets in the horizontal direction.

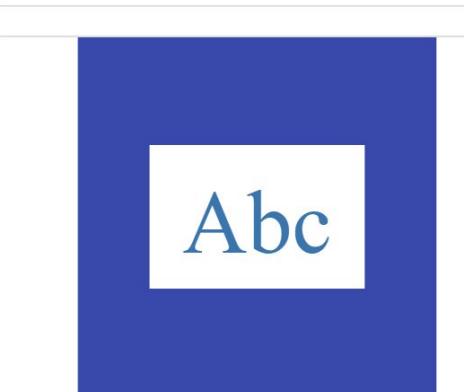
# Widgets

## São componentes do Flutter



### Scaffold

Implements the basic Material Design visual layout structure. This class provides APIs for showing drawers, snack bars, and bottom sheets.



### Text

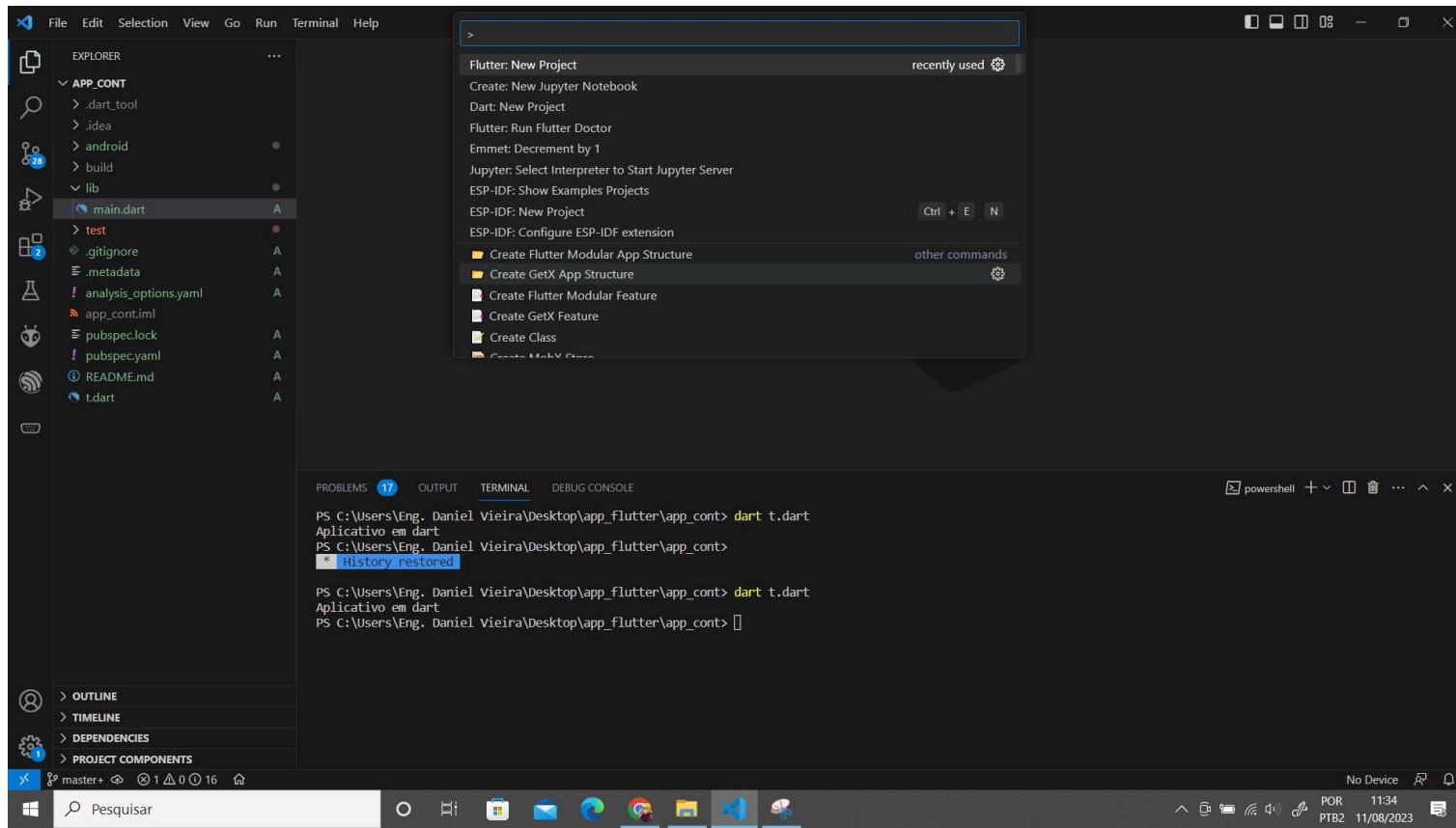
A run of text with a single style.

# Criando projeto Flutter no VSCode

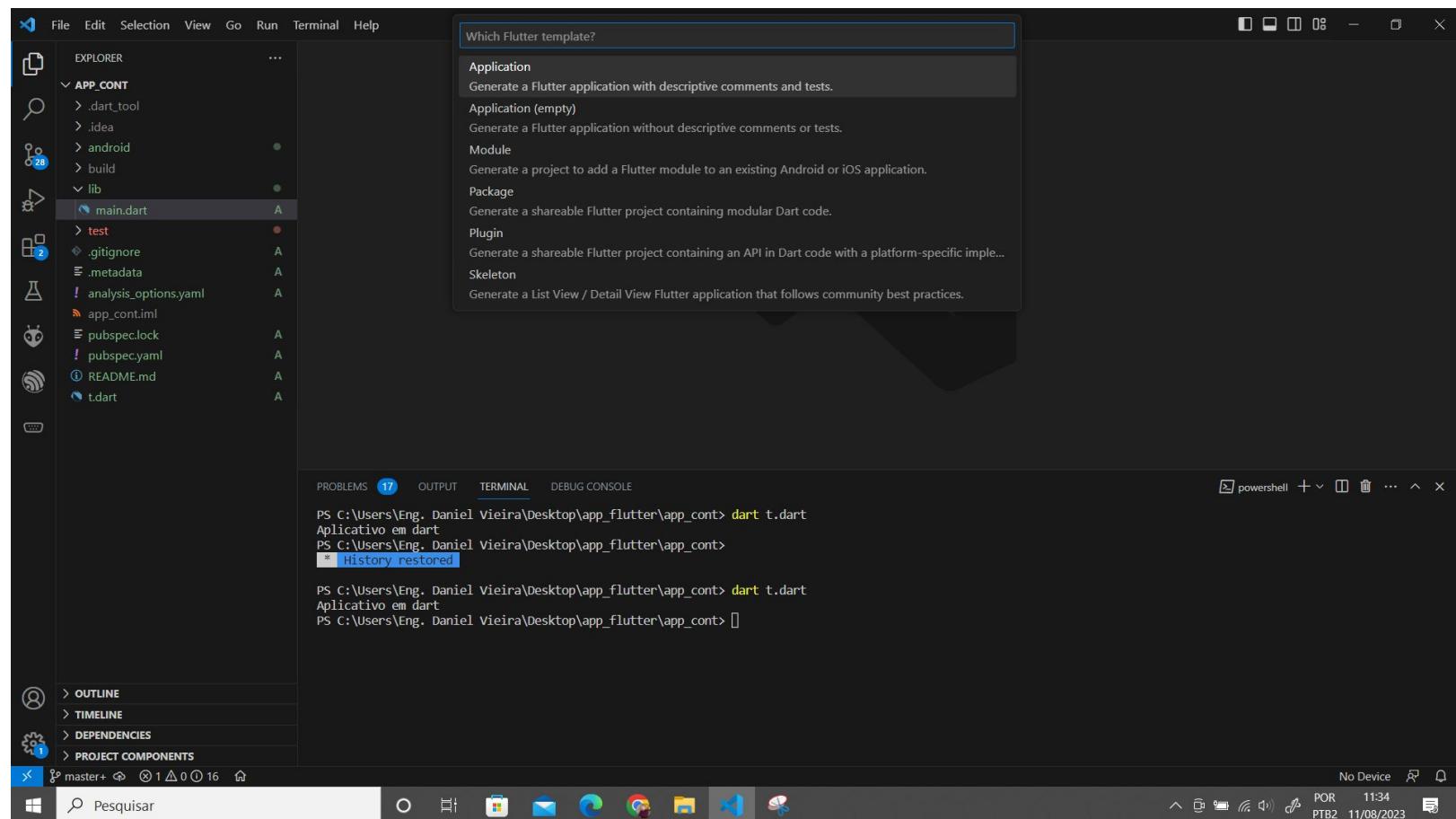
Abrir o VSCODE e apertar a tecla F1

Após apertar F1 clicar em Flutter New Project

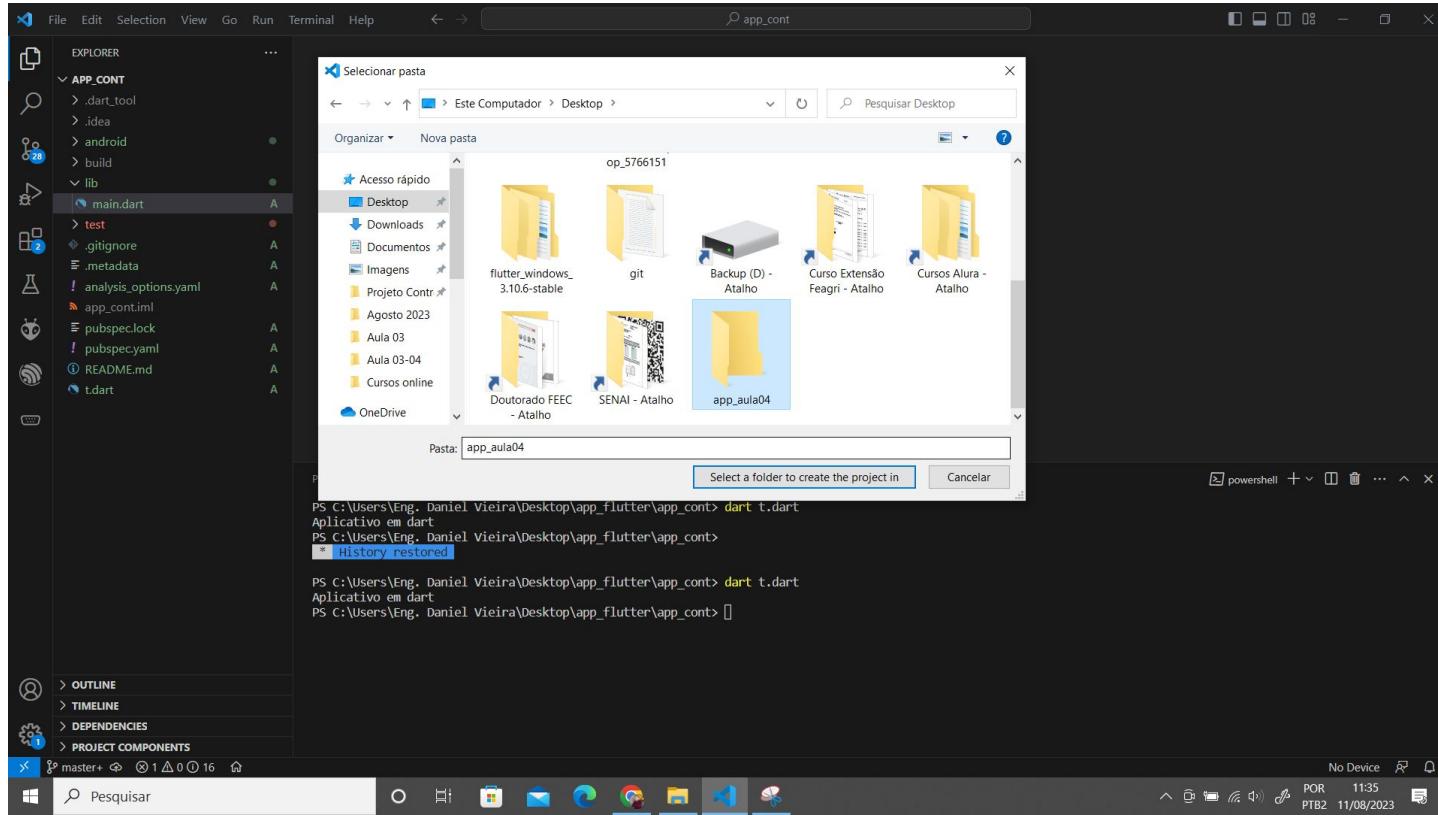
# Criando projeto Flutter no VSCode



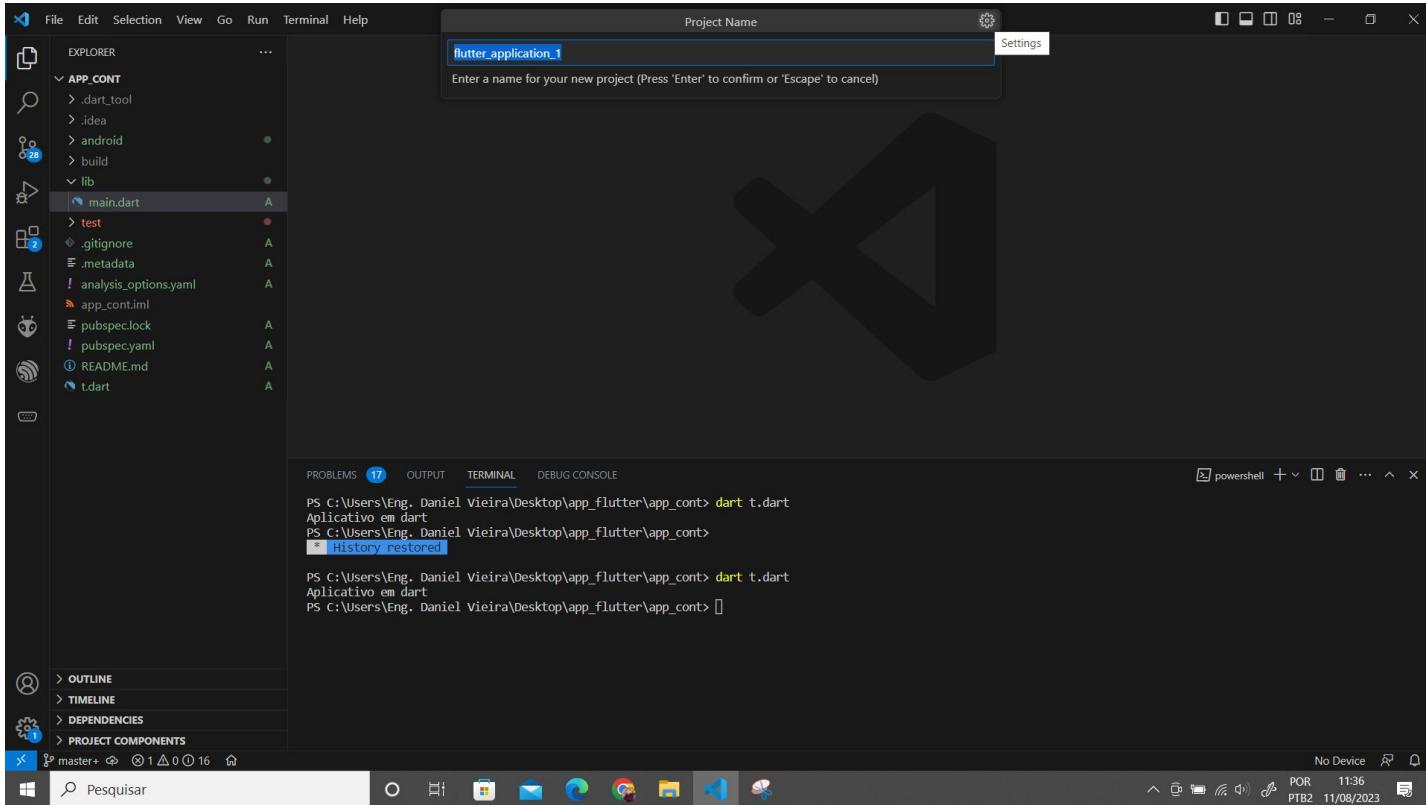
# Criando projeto Flutter no VSCode



# Criando projeto Flutter no VSCode

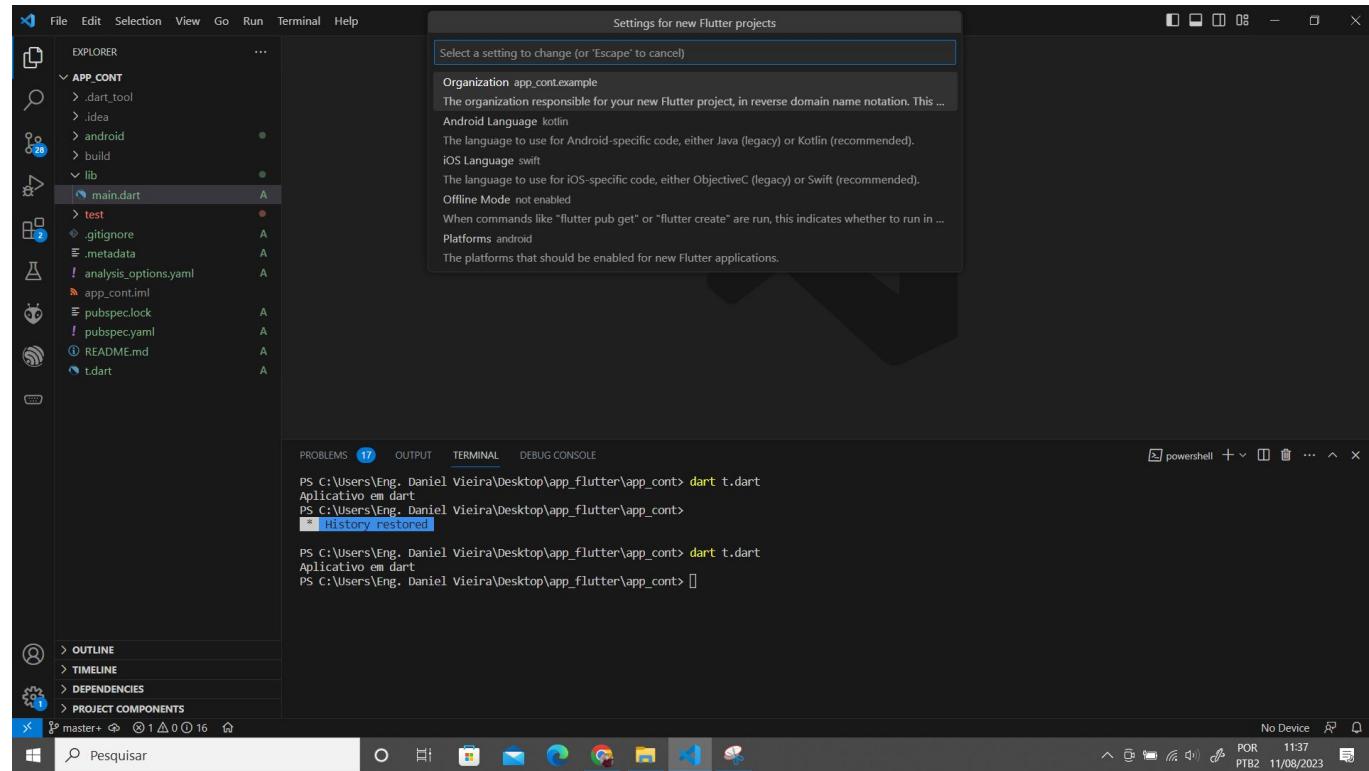


# Criando projeto Flutter no VSCode



# Criando projeto Flutter no VSCode

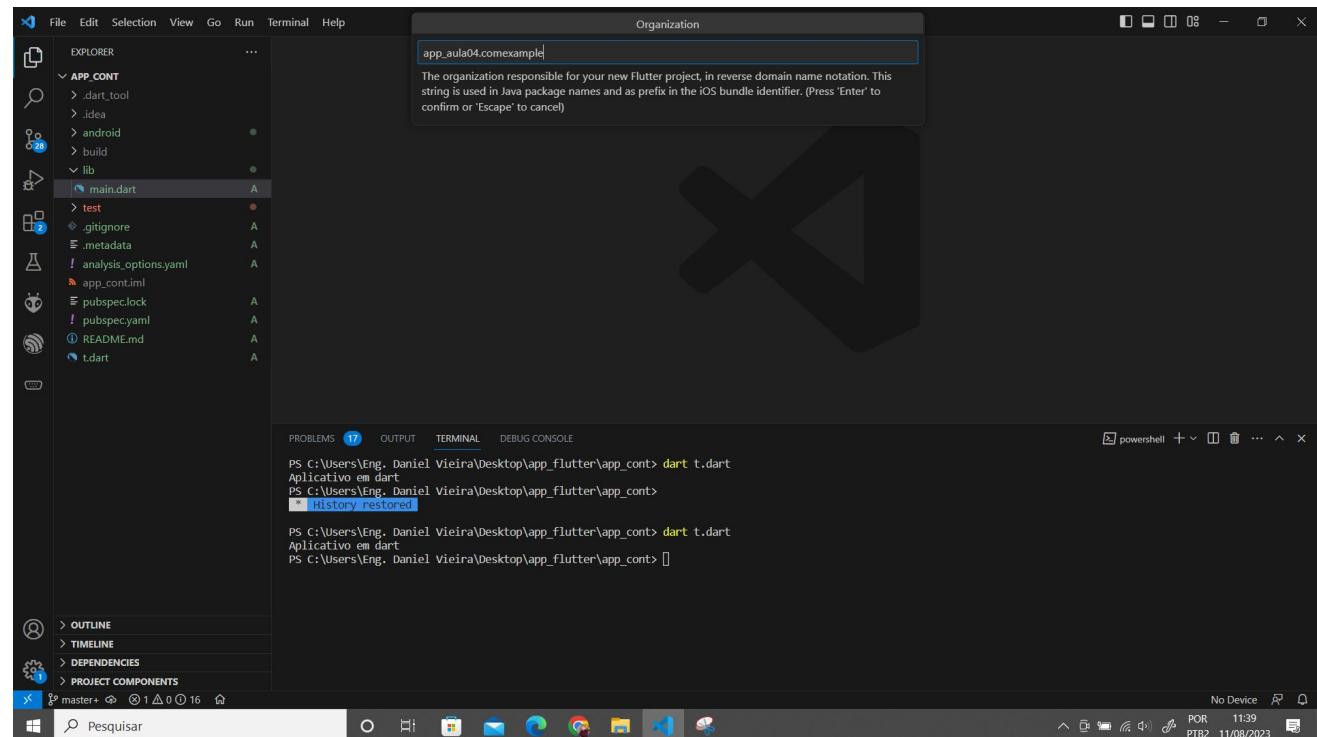
Parâmetros de configuração para criar o app



# Criando projeto Flutter no VSCode

Organization é o parâmetro que indica o domínio da empresa que desenvolveu o app.

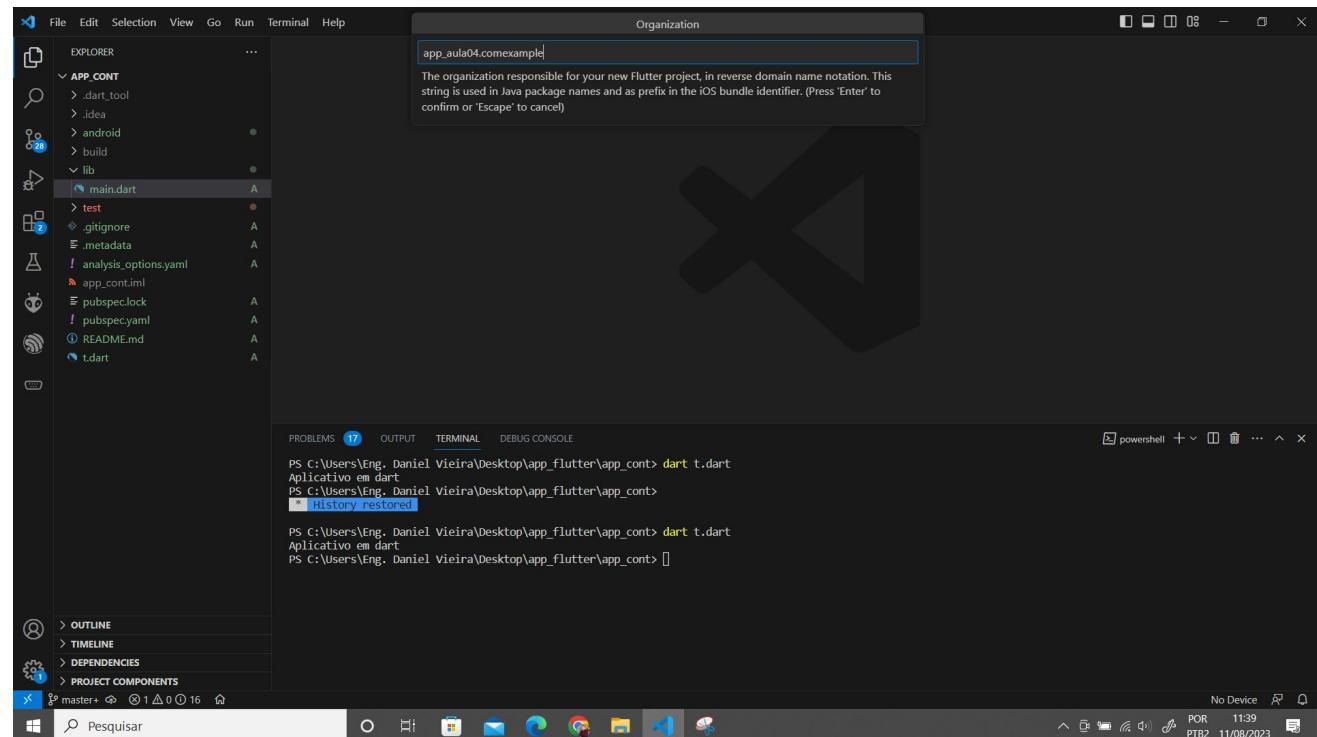
É um parâmetro importante para publicação do APP em lojas virtuais



# Criando projeto Flutter no VSCode

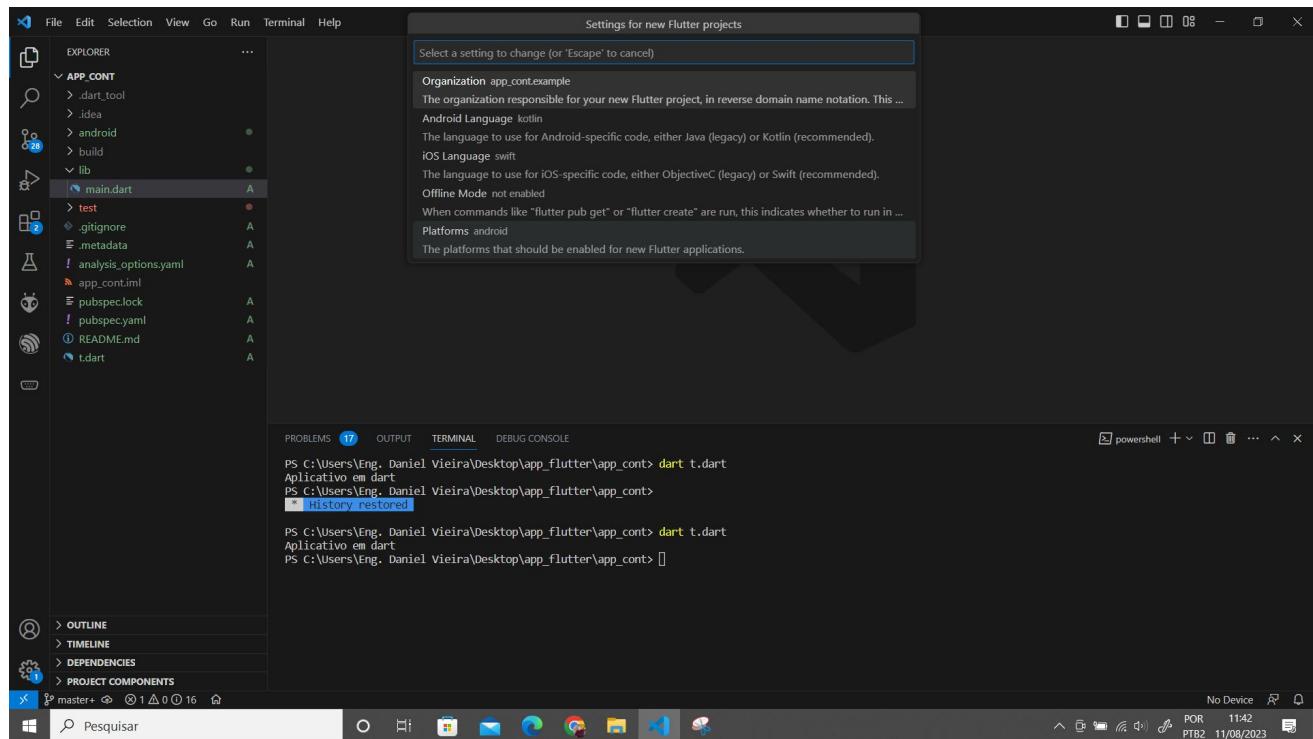
Organization é o parâmetro que indica o domínio da empresa que desenvolveu o app.

É um parâmetro importante para publicação do APP em lojas virtuais



# Criando projeto Flutter no VSCode

Platforms é possível selecionar as plataformas que o app será desenvolvido



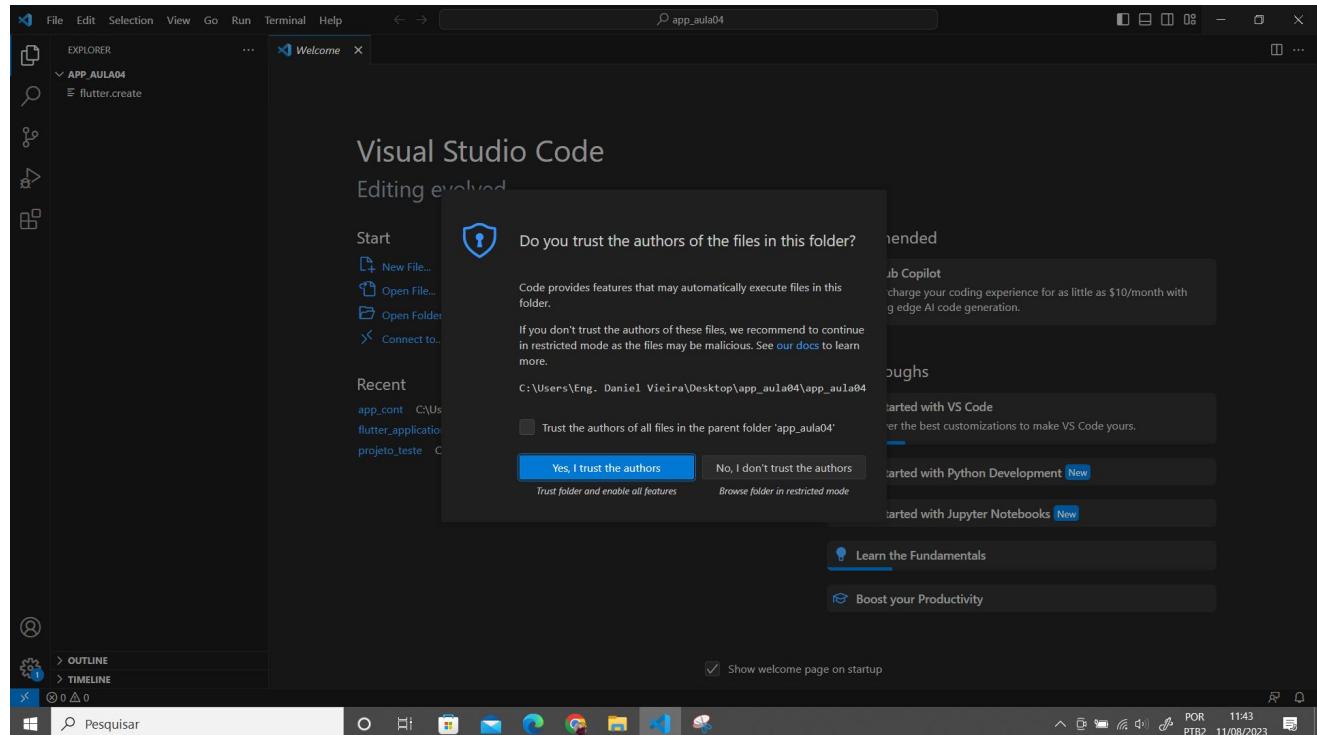
# Criando projeto Flutter no VSCode

Após realizar as configurações

Apertar Esc e enter  
O projeto Flutter será criado

Marcar a opção

Yes, I Trust the authors



# Criando projeto Flutter no VSCode

Escolhendo o emulador  
para emular o telefone  
Clicar em no device

The screenshot shows the Visual Studio Code interface with a Flutter project named 'APP\_AULA04'. The Explorer sidebar on the left displays the project files, including 'main.dart' which is currently selected. The main code editor area shows the Dart code for a simple Flutter application. The status bar at the bottom provides information about the file (ln 12, Col 39), encoding (UTF-8), and other settings like 'Dart' and 'No Device'. The taskbar at the bottom includes icons for file operations and system notifications.

```
File Edit Selection View Go Run Terminal Help ↵ → 🔍 app_aula04
EXPLORER APP_AULA04
lib/main.dart
lib/main.dart > MyApp > build
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9
10 // This widget is the root of your application.
11 @override
12 Widget build(BuildContext context) {
13   return MaterialApp(
14     title: 'Flutter Demo',
15     theme: ThemeData(
16       // This is the theme of your application.
17       //
18       // TRY THIS: Try running your application with "flutter run". You'll see
19       // the application has a blue toolbar. Then, without quitting the app,
20       // try changing the seedColor in the colorScheme below to Colors.green
21       // and then invoke "hot reload" (save your changes or press the "hot
22       // reload" button in a Flutter-supported IDE, or press "r" if you used
23       // the command line to start the app).
24       //
25       // Notice that the counter didn't reset back to zero; the application
26       // state is not lost during the reload. To reset the state, use hot
27       // restart instead.
28       //
29       // This works for code too, not just values: Most code changes can be
Run | Debug | Profile
```

# Criando projeto Flutter no VSCode

Após escolher o emulador  
ele irá abrir na tela



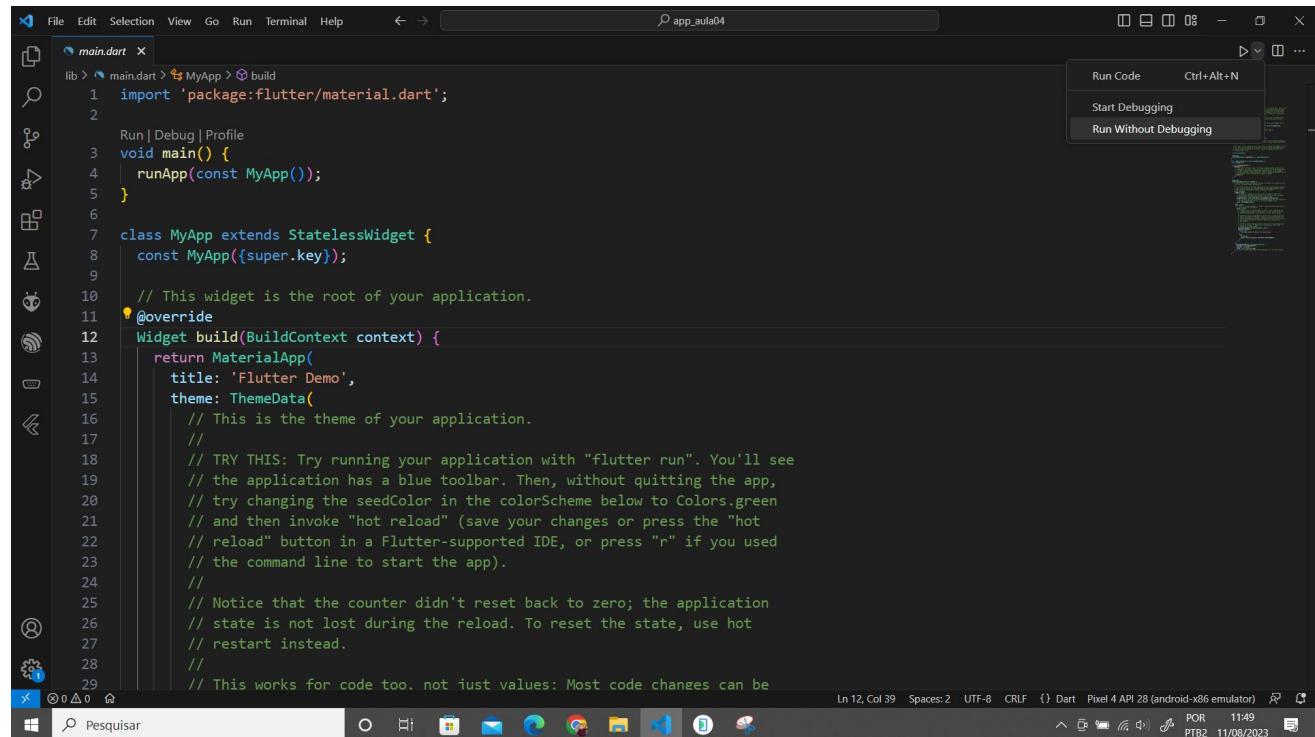
```
main.dart
import 'package:flutter/material.dart';
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        // This is the theme of your application.
        //
        // TRY THIS: Try running your application with "flutter run". You'll see
        // the application has a blue toolbar. Then, without quitting the app,
        // try changing the seedColor in the colorScheme below to Colors.green
        // and then invoke "hot reload" (save your changes or press the "hot
        // reload" button in a Flutter-supported IDE, or press "r" if you used
        // the command line to start the app).
        //
        // Notice that the counter didn't reset back to zero; the application
        // state is not lost during the reload. To reset the state, use hot
        // restart instead.
        //
        // This works for code too. not just values: Most code changes can be
      ),
    );
  }
}
```

# Criando projeto Flutter no VSCode

Para executar o APP  
clicar em Run without  
debug



```
lib/main.dart
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9
10 // This widget is the root of your application.
11 @override
12 Widget build(BuildContext context) {
13   return MaterialApp(
14     title: 'Flutter Demo',
15     theme: ThemeData(
16       // This is the theme of your application.
17       //
18       // TRY THIS: Try running your application with "flutter run". You'll see
19       // the application has a blue toolbar. Then, without quitting the app,
20       // try changing the seedColor in the colorsScheme below to Colors.green
21       // and then invoke "hot reload" (save your changes or press the "hot
22       // reload" button in a Flutter-supported IDE, or press "r" if you used
23       // the command line to start the app).
24       //
25       // Notice that the counter didn't reset back to zero; the application
26       // state is not lost during the reload. To reset the state, use hot
27       // restart instead.
28       //
29       // This works for code too. not just values: Most code changes can be
```

Ln 12, Col 39 Spaces:2 UTF-8 CRLF Dart Pixel 4 API 28 (android-x86 emulator)

Pesquisar

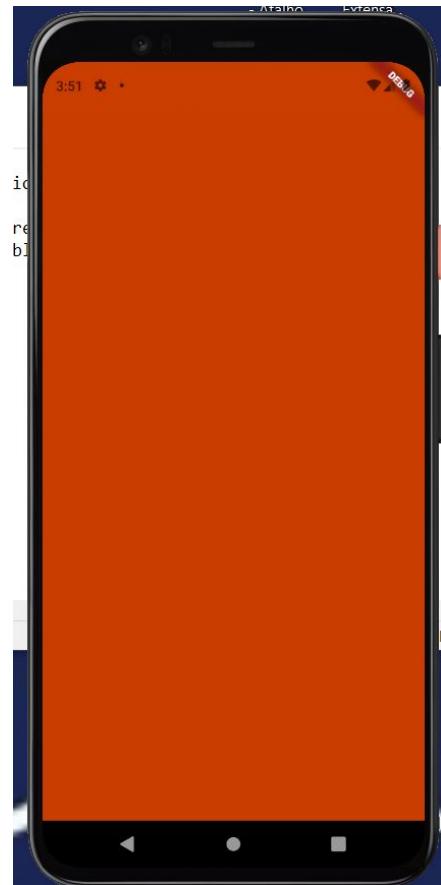
Run Code Ctrl+Alt+N

Start Debugging

Run Without Debugging

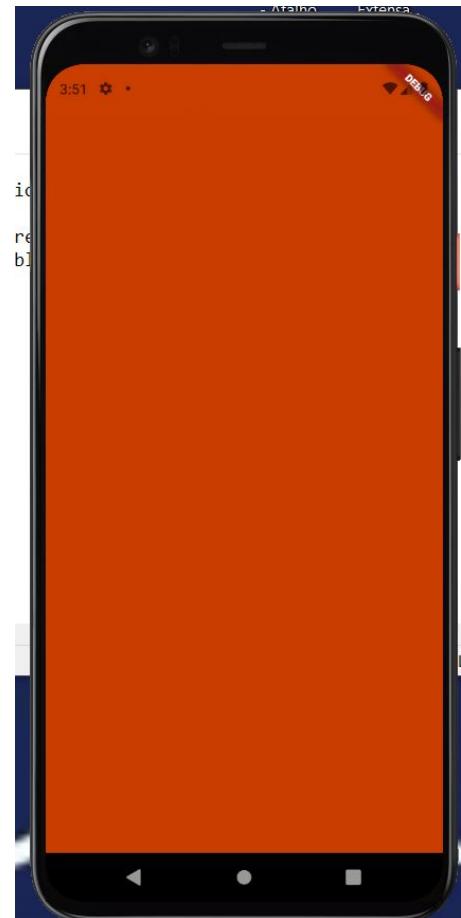
# Container Flutter

```
home: Container(color: Color.red) // Cria um container com a cor vermelha,
```



# Container Flutter

```
home: Container(color: Color.red), cria um container da cor vermelha  
    Container(color:Color.blue),// Cria um container da cor azul  
 );  
 }
```



# Container e Stack Flutter

home:

```
// Componente Stack permite criar um Container dentro do outro
Stack(alignment: AlignmentDirectional.center // Alinhamento no centro do Container
,children: // Estabelece a relação entre os Widgets Pais e filhos
[
    Container(color: Colors.red, width: 1000, height: 1000),// Dimensão e cor do Container
    Container(color: Colors.blue, width:500,height:500),
    Text("APP Flutter",style: TextStyle(color:Colors.red) ), // Cria texto no APP
],
)
);
}
}
```



# Container e Stack Flutter

home:

```
Stack(alignment: AlignmentDirectional.center,  
children:  
[  
    Container(color:Colors.red,width:1000,height:1000),  
    Container(color:Colors.blue,width:500,height:500),  
    Text('APP Flutter', style: TextStyle(color:Colors.red,  
decoration: TextDecoration.none,)),  
],  
);  
}  
}
```



# Container e Stack Flutter

home:

```
Stack(alignment: AlignmentDirectional.center,  
children:  
[  
    Container(color:Colors.red,width:1000,height:1000),  
    Container(color:Colors.blue,width:500,height:500),  
    Text('APP Flutter', style: TextStyle(color:Colors.red,  
    fontSize:50,decoration: TextDecoration.none,)),  
],  
));  
}  
}
```



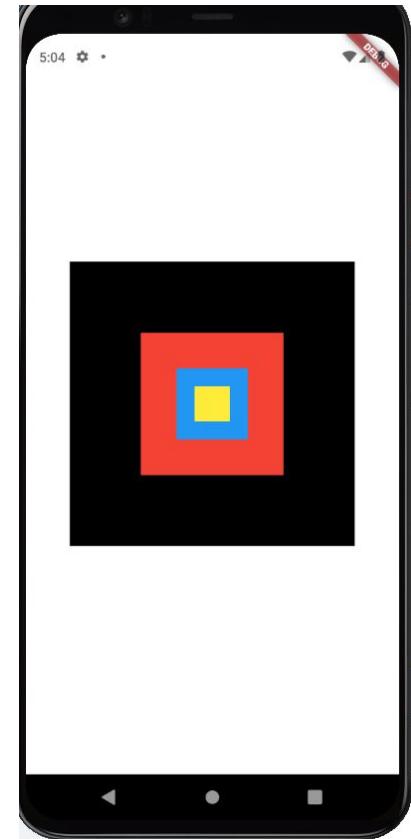
# Container e Stack Flutter

```
import 'package:flutter/material.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

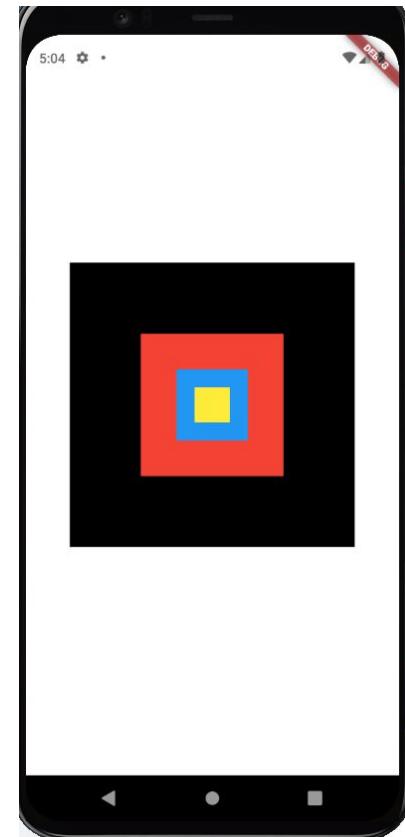
  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.blue),
        useMaterial3: true,
    ),
}
```



# Container e Stack Flutter

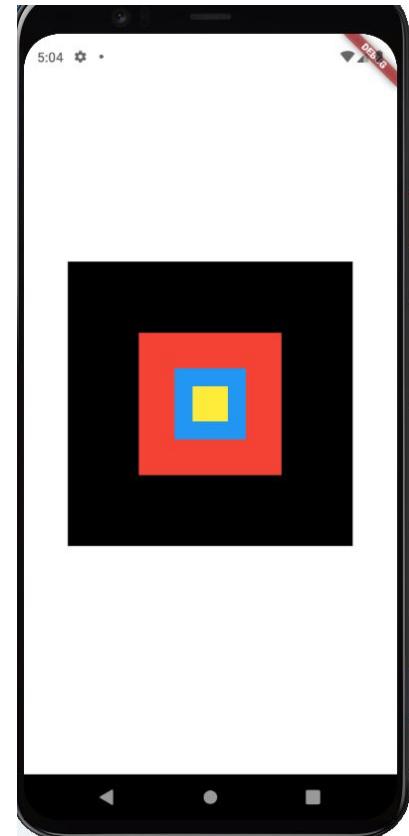
home:

```
Stack(alignment: AlignmentDirectional.center,  
children: [  
    Container(color: Colors.white,width:800,height:800),  
    Container(color: Colors.black,width:300,height:300),  
    Container(color: Colors.red,width:150,height:150),  
    Container(color: Colors.blue,width:75,height:75),  
    Container(color: Colors.yellow,width:37,height:37),  
],  
);  
}  
}
```



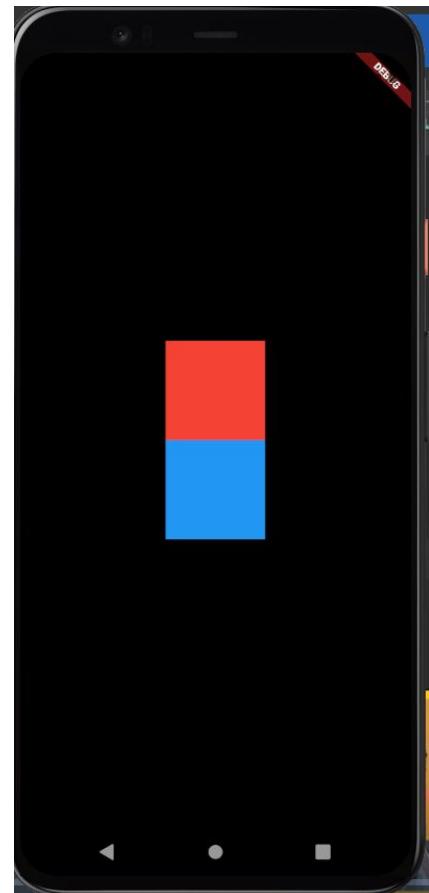
# Container e Stack Flutter

```
Stack(alignment: AlignmentDirectional.center,  
      children: [  
        Container(color: Colors.white,width:800,height:800),  
        Container(color: Colors.black,width:300,height:300),  
        Container(color: Colors.red,width:150,height:150),  
        Container(color: Colors.blue,width:75,height:75),  
        Container(color: Colors.yellow,width:37,height:37),  
      ],  
    );  
}  
}
```



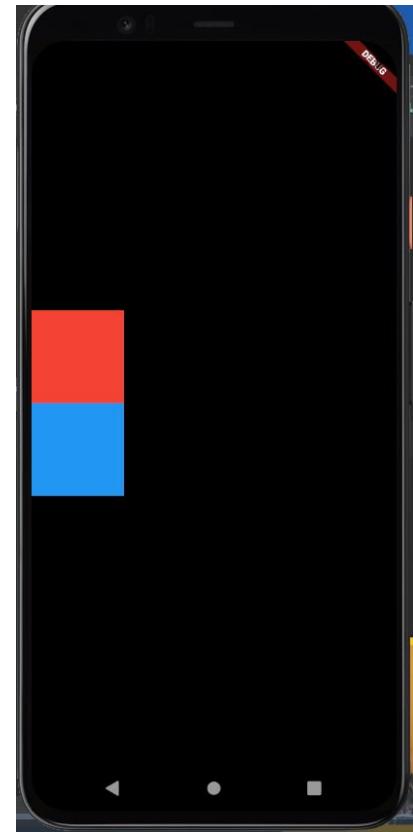
# Widget Column e Row

```
home: Column(  
    mainAxisAlignment: MainAxisAlignment.center, // Alinhamento na vertical  
    children: [ // herança pais e filhos  
        Container(color:Colors.red,width:100,height:100),  
        Container(color:Colors.blue,width:100,height:100),  
    ],  
)
```



# Widget Column e Row

```
home: Column(  
    mainAxisAlignment: MainAxisAlignment.center, // Alinhamento na vertical  
    crossAxisAlignment: CrossAxisAlignment.start, // Alinhamento no eixo  
    Secundário  
    children: [  
        Container(color:Colors.red,width:100,height:100),  
        Container(color:Colors.blue,width:100,height:100),  
    ],  
)
```



# Widget Column e Row

Row(

```
    mainAxisAlignment: MainAxisAlignment.center, // alinhamento horizontal  
    crossAxisAlignment: CrossAxisAlignment.center, // alinhamento no eixo
```

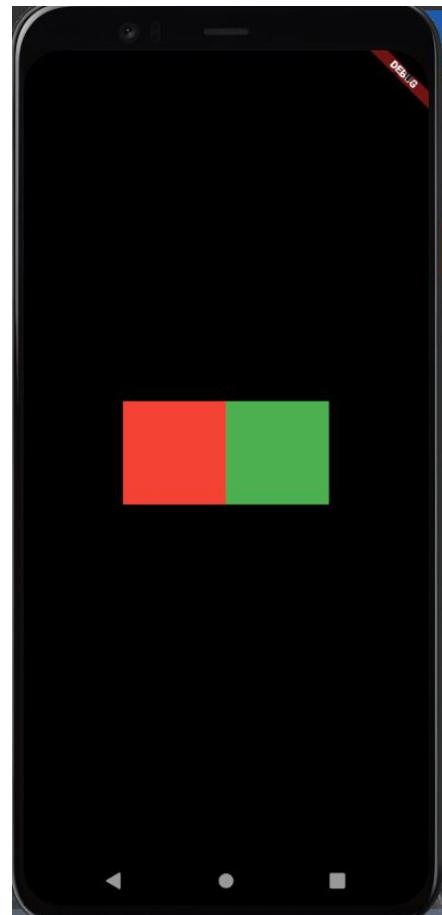
vertical

```
    children: [
```

```
        // herança dos widgets
```

```
        Container(color:Colors.red, width:100, height:100),  
        Container(color:Colors.green, width:100,height:100),
```

```
    ],),
```



# Criando Tela

home:

```
Column(  
    mainAxisAlignment: MainAxisAlignment.spaceEvenly, // alinhamento vertical  
    crossAxisAlignment: CrossAxisAlignment.center, // alinhamento horizontal  
    children:[ // herança  
        Stack(  
            alignment: AlignmentDirectional.center, // alinhamento do empilhamento  
            children: [  
                Container(color:Colors.red, width:100,height:100),  
                Container(color:Colors.blue,width:50, height:50),  
            ],),  
        ], // children 1  
    ),  
);  
}  
}
```

# Criando Tela

```
Stack(  
    alignment: AlignmentDirectional.center, // alinhamento do empilhamento  
    children: [  
        Container(color:Colors.red, width:100,height:100),  
        Container(color:Colors.blue,width:50, height:50),  
    ],  
    ], // children 1
```

# Criando Tela

Row(

    mainAxisAlignment: MainAxisAlignment.spaceEvenly,

    crossAxisAlignment: CrossAxisAlignment.center,

    children: [

        Container(color:Colors.blue, height:50,width:50),

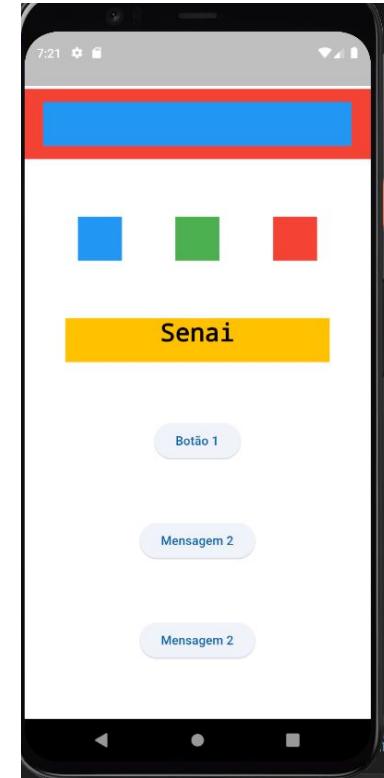
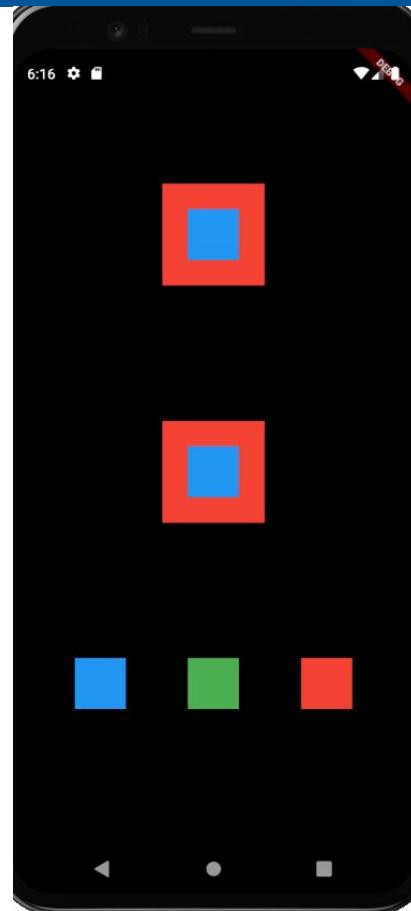
        Container(color:Colors.green, height:50,width:50),

        Container(color:Colors.red, height:50,width:50),

    ],

)

# Criando Tela



# Código

```
import 'package:flutter/material.dart';
int cont =0;
void _msg()
{
    cont = cont+1;
    print("Desenvolvimento Mobile 1");
    print("Contagem $cont");
}
void _msg2()
{
    cont = cont-1;
    print("Contagem $cont");
    print("Senai");
}
void main() {
    runApp(const MyApp());
    // remove a faixa debug do app
}
```

# Código

```
class MyApp extends StatelessWidget {  
  const MyApp({super.key});  
  
  // This widget is the root of your application.  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      debugShowCheckedModeBanner: false, // Remove a faixa debug do app  
      title: 'Flutter Demo',  
      theme: ThemeData(  
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.blue),  
        useMaterial3: true,  
      ),  
      home: Container(  
        color: Colors.white,  
        child: Column(  
          mainAxisSize: MainAxisSize.spaceEvenly,  
          children: [  
            Stack(  
              alignment: AlignmentDirectional.center,  
              children: [  
                Container(color: Colors.red, width: 400, height: 80),  
                Container(color: Colors.blue, width: 350, height: 50,),  
              ],  
            ),  
          ],  
        ),  
      ),  
    );  
  }  
}
```

# Código

Row(

```
    mainAxisAlignment: MainAxisAlignment.spaceEvenly,  
    crossAxisAlignment: CrossAxisAlignment.end,  
    children: [  
        Container(color: Colors.blue, height: 50, width: 50),  
        Container(color: Colors.green, height: 50, width: 50),  
        Container(color: Colors.red, height: 50, width: 50),  
    ],  
,  
    Container(color: Colors.amber, height: 50, width: 300,  
    child: Text("Senai", style: TextStyle(color: Colors.black, fontSize: 28,  
    decoration: TextDecoration.none), textAlign: TextAlign.center,),  
,
```

# Código

```
ElevatedButton(  
    onPressed: (){  
        print("Você apertou o botao1");  
    },  
    child: Text("Botão 1"),),  
ElevatedButton(  
    onPressed: _msg,  
    child: Text("Mensagem 2")),  
  
ElevatedButton(  
onPressed: _msg2,  
child: Text("Mensagem 2")),  
],  
(  
),  
(  
);  
}  
}
```

# Obrigado!

Prof. Me Daniel Vieira

Email: [danielvieira2006@gmail.com](mailto:danielvieira2006@gmail.com)

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95

