



Flutter

Wesley Dias Maciel

2021/02



Prática 04

Row

Documentação: <https://api.flutter.dev/flutter/widgets/Row-class.html>

Objetivo: apresentar o widget básico Row. O widget Row permite criar layouts flexíveis na direção horizontal (linha). Os objetos filhos de Row são posicionados no eixo x do plano cartesiano. Objetos da classe Row são baseados no modelo de layout flexbox da Web.

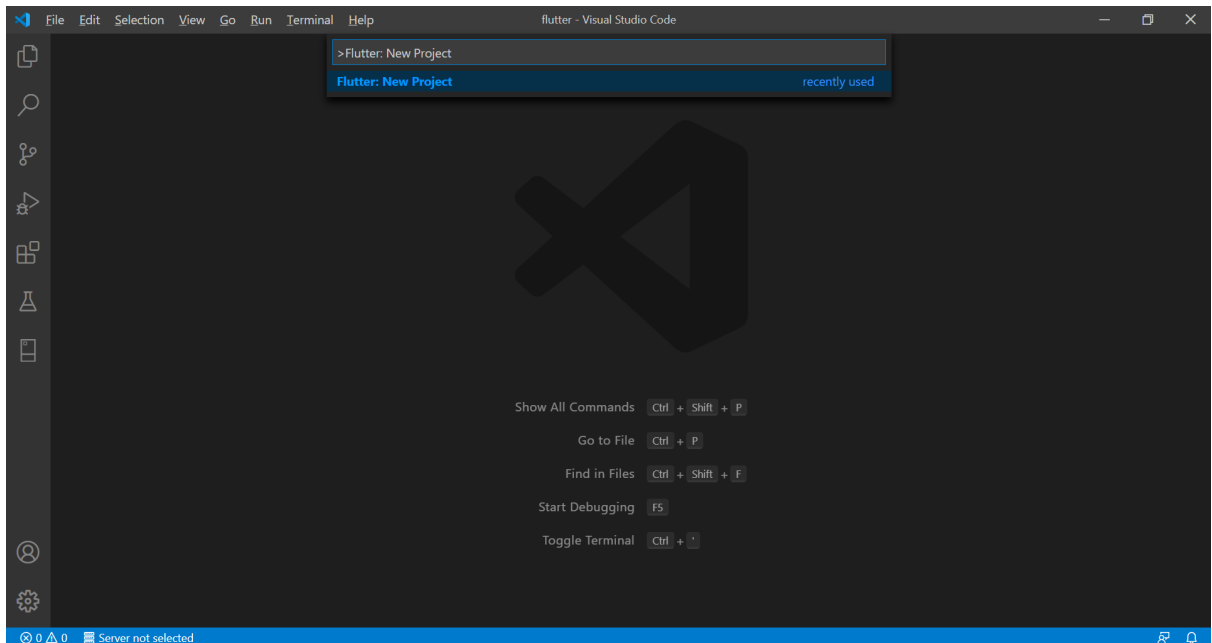
O widget Row exibe seus filhos em um vetor horizontal. Para que um filho do widget Row se expanda e preencha todo o espaço horizontal disponível, é preciso envolvê-lo em um widget Expanded. O widget Row não possui a funcionalidade de rolagem (scroll).

Em geral, é considerado um erro que o widget Row possua um ou mais filhos que não caibam no espaço disponível na tela do dispositivo. Caso você tenha widgets dispostos em linha e queira que eles possam rolar, se não houver espaço suficiente na linha, então, use um widget ListView.

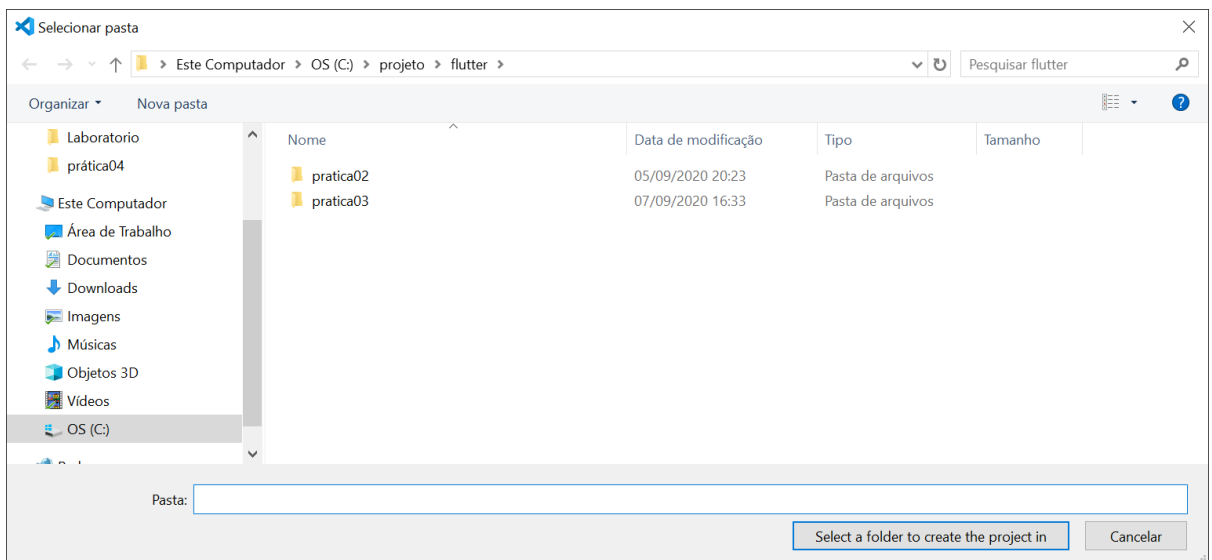
Para dispor widgets na direção vertical, use o widget Column. Por outro lado, se você tem apenas um widget filho, use os widget Align ou Center para posicioná-lo.

- 1) No Visual Studio Code, crie um novo projeto Flutter. Clique em View > Command Palette ou pressione CTRL + SHIFT + P. Na caixa de entrada, informe:

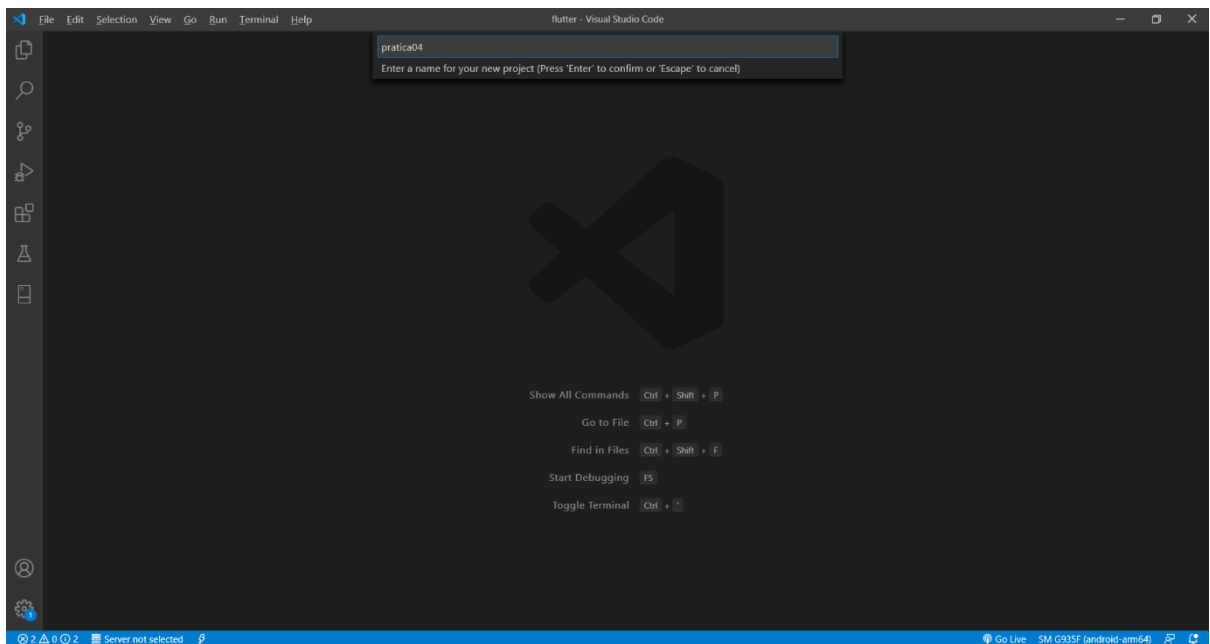
>Flutter: New Project



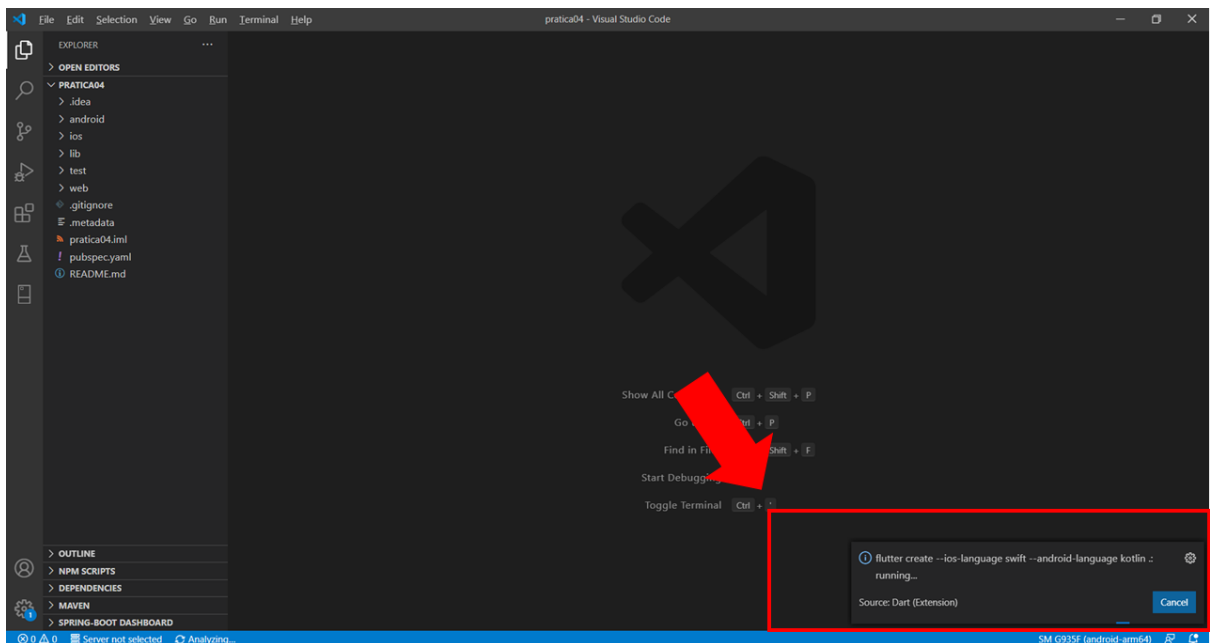
- 2) Na nova janela aberta, selecione um diretório para seu projeto. **OBS:** o caminho do diretório deve ter apenas caracteres ASCII. Além disso, o caminho do diretório não pode ter espaços em branco. Dessa forma, não use: acentos, símbolos, espaços, cedilha, etc. Exemplo: C:\projeto\flutter.



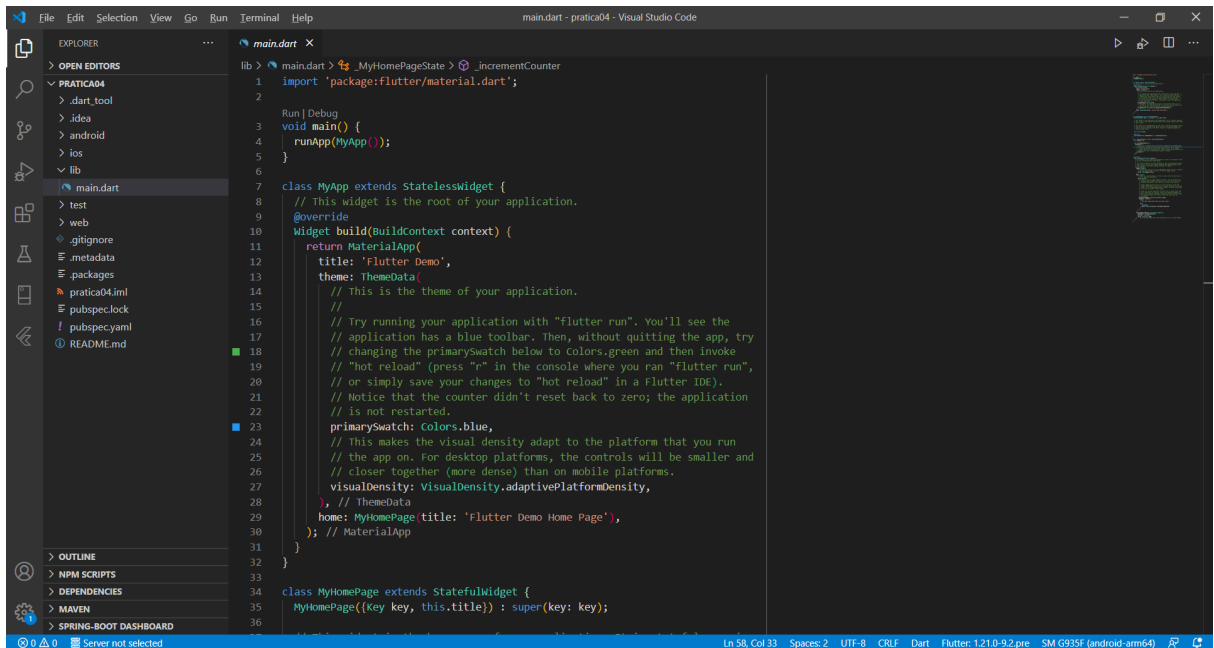
- 3) Na nova caixa de entrada, informe um nome para seu projeto. Exemplo: pratica04. Em seguida, pressione a tecla ENTER. **OBS:** o nome do projeto deve ter apenas caracteres ASCII (<https://pt.wikipedia.org/wiki/ASCII>). Além disso, o nome do projeto não pode ter espaços em branco. Dessa forma, não use: acentos, símbolos, espaços, cedilha, etc.



4) Aguarde a finalização do processo de criação de seu projeto.



5) Após a criação de seu projeto, observe a estrutura de diretórios criada e o código inicial gerado como exemplo em lib/main.dart.

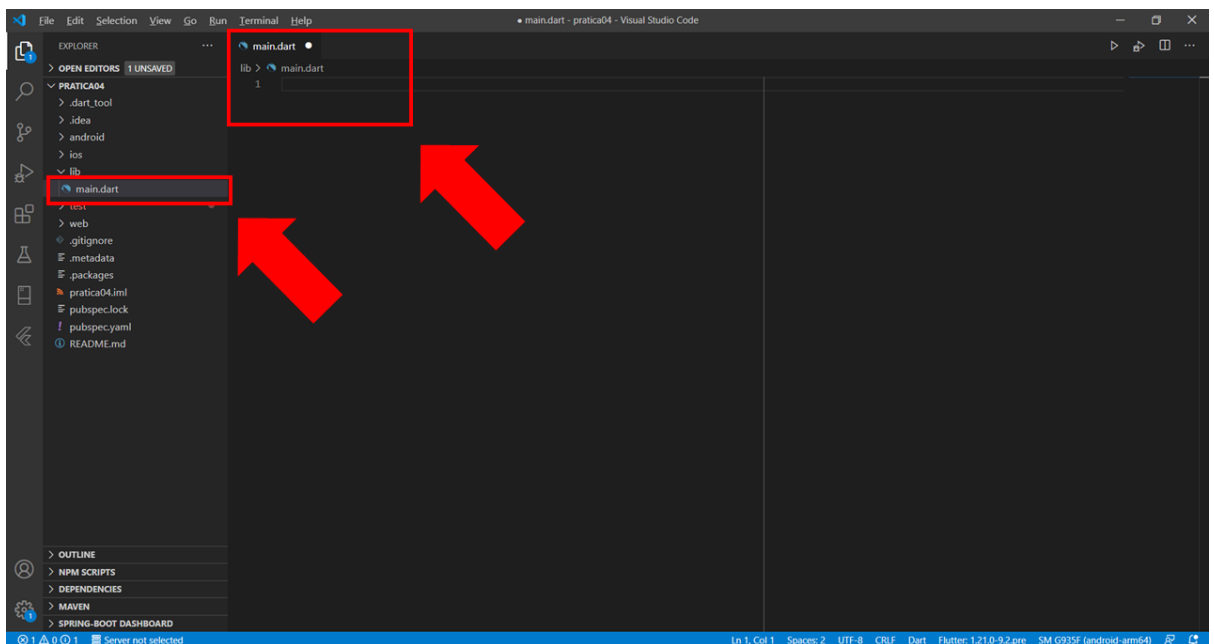


```

lib > main.dart > _MyHomePageState > incrementCounter
1  import 'package:flutter/material.dart';
2
3  Run | Debug
4  void main() {
5    runApp(MyApp());
6  }
7
8  class MyApp extends StatelessWidget {
9    // This widget is the root of your application.
10   @override
11   Widget build(BuildContext context) {
12     return MaterialApp(
13       title: 'Flutter Demo',
14       theme: ThemeData(
15         // This is the theme of your application.
16         // Try running your application with "flutter run". You'll see the
17         // application has a blue toolbar. Then, without quitting the app, try
18         // changing the primarySwatch below to Colors.green and then invoke
19         // "hot reload" (press "r" in the console where you ran "flutter run",
20         // or simply save your changes to "hot reload" in a Flutter IDE).
21         // Notice that the counter didn't reset back to zero; the application
22         // is not restarted.
23         primarySwatch: Colors.blue,
24         // This makes the visual density adapt to the platform that you run
25         // the app on. For desktop platforms, the controls will be smaller and
26         // closer together (more dense) than on mobile platforms.
27         visualDensity: VisualDensity.adaptivePlatformDensity,
28       ), // ThemeData
29       home: MyHomePage(title: 'Flutter Demo Home Page'),
30     ); // MaterialApp
31   }
32 }
33
34 class MyHomePage extends StatefulWidget {
35   MyHomePage({Key key, this.title}) : super(key: key);
36

```

6) Apague o código inicial gerado como exemplo em lib\main.dart.



```

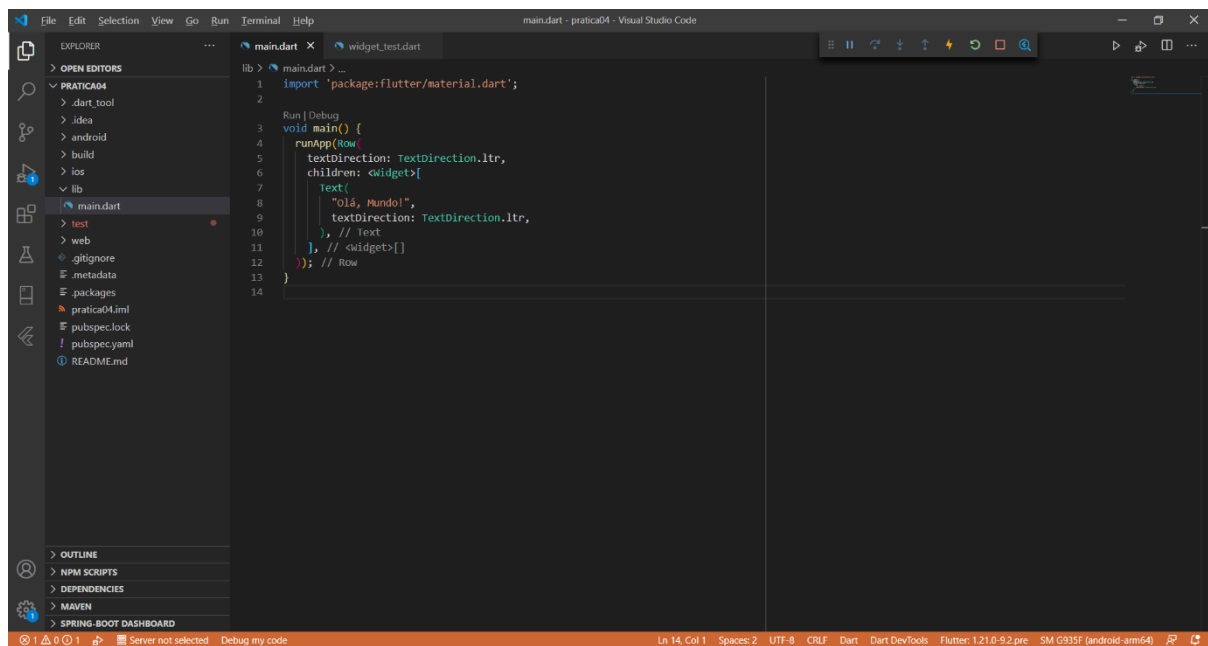
lib > main.dart
1  main.dart

```

- 7) Um aplicativo Flutter mínimo possui uma função principal main() que simplesmente chama a função runApp() com um widget.
- 8) No exemplo abaixo, a função runApp() é chamada com um widget para linha (Row). O aplicativo apresenta na tela um texto.

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

void main() {
  runApp(Row(
    textDirection: TextDirection.ltr,
    children: [
      Text(
        "Olá, Mundo!",
        textDirection: TextDirection.ltr,
      ),
    ],
  ));
}
```



- 9) Pressione a tecla F5, para iniciar a depuração, debug, do código. Aguarde a finalização do processamento do Gradle. Na janela do Google Chrome ou na tela do Vysor (<https://vysor.io/>), observe a saída gerada.



OBS:

Caso o framework Flutter dispare uma mensagem de erro no arquivo de teste, **test/main.dart**, apague todo o conteúdo do arquivo de teste.

- 10) No widget Row, o eixo principal (**main axis**) é o eixo horizontal, eixo x. Por outro lado, o eixo transversal (**cross axis**) é o eixo vertical, eixo y. Dessa forma, os parâmetros `mainAxisAlignment` e `mainAxisSize` aplicam configuração de alinhamento e tamanho, respectivamente, ao eixo horizontal, eixo x. Analogamente, os parâmetros `crossAxisAlignment` e `crossAxisSize` aplicam configuração de alinhamento e tamanho, respectivamente, ao eixo vertical, eixo y. **OBS:** no caso do widget Column, essa lógica se inverte: o eixo principal (**main axis**) é o eixo vertical, eixo y, e o eixo transversal (**cross axis**) é o eixo horizontal, eixo x. Exemplo:

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

void main() {
  runApp(
    Row(
      mainAxisAlignment: MainAxisAlignment.end,
      textDirection: TextDirection.ltr,
      children: [
        Text(
          "Olá, Mundo!",
          textDirection: TextDirection.ltr,
        ),
      ],
    ),
  );
}
```



11) Altere o código, incluindo formatação ao texto.


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

void main() {
  runApp(Row(
    textDirection: TextDirection.ltr,
    children: [
      Text(
        "Olá, Mundo!",
        textDirection: TextDirection.ltr,
        style: TextStyle(
          color: Colors.green,
          fontSize: 30,
          fontWeight: FontWeight.bold,
          backgroundColor: Colors.white,
        ), //TextStyle.
      ),
    ],
  ));
}
```

- 12) Clique em File > Save ou pressione CTRL + S para salvar a alteração. Caso necessário, clique no botão de Hot Reload ou no botão de Restart. Veja a alteração numa janela do Google Chrome ou numa tela do Vysor:



13) Altere o código, aumentando o tamanho do texto.

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

void main() {
  runApp(Row(
    textDirection: TextDirection.ltr,
    children: [
      Text(
        "Olá, Mundo! Este é o meu primeiro aplicativo usando o widget Row. O w",
        textDirection: TextDirection.ltr,
        style: TextStyle(
          color: Colors.green,
          fontSize: 30,
          fontWeight: FontWeight.bold,
          backgroundColor: Colors.white,
```

```
    ), //TextStyle.  
  ),  
],  
));  
}
```



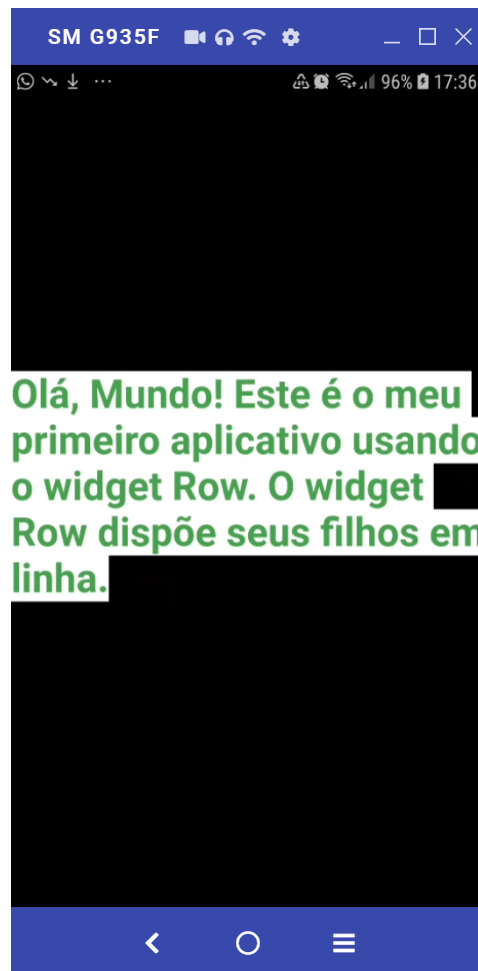
Observe que não foi possível alocar espaço suficiente na tela do dispositivo para esse tamanho de texto. O framework indica esse estouro do espaço disponível, apresentando uma tarja lateral. Nesse caso, será necessário usar um widget Expanded.

14) Altere o código, incluindo um widget Expanded, para que o texto seja adaptado ao tamanho da tela.

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(Row(  
    textDirection: TextDirection.ltr,  
    children: [  
      Expanded(  
        child: Text(  

```

```
"Olá, Mundo! Este é o meu primeiro aplicativo usando o widget Row. O  
widget Row dispõe seus filhos em linha.",  
  textDirection: TextDirection.ltr,  
  style: TextStyle(  
    color: Colors.green,  
    fontSize: 30,  
    fontWeight: FontWeight.bold,  
    backgroundColor: Colors.white,  
  ), //TextStyle.  
),  
),  
],  
));  
}
```



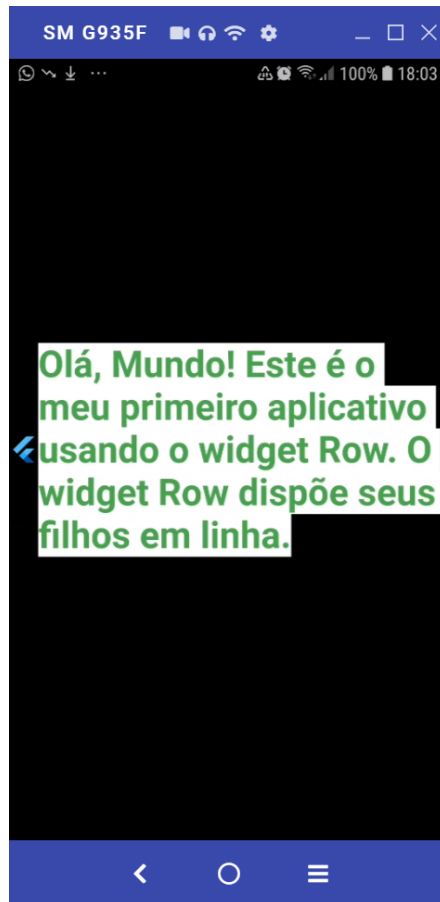
15) Altere o código novamente, incluindo o logo do Flutter.



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

void main() {
  runApp(Row(
    textDirection: TextDirection.ltr,
    children: [
      FlutterLogo(),
      Expanded(
        child: Text(
          "Olá, Mundo! Este é o meu primeiro aplicativo usando o widget Row. O  

          widget Row dispõe seus filhos em linha.",
          textDirection: TextDirection.ltr,
          style: TextStyle(
            color: Colors.green,
            fontSize: 30,
            fontWeight: FontWeight.bold,
            backgroundColor: Colors.white,
          ), //TextStyle.
        ),
      ),
    ],
  ));
}
```

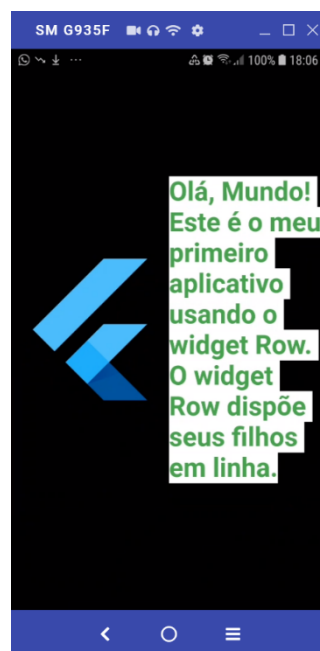


16) Altere o código novamente, como indicado abaixo.

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

void main() {
  runApp(Row(
    textDirection: TextDirection.ltr,
    children: [
      Expanded(
        child: FittedBox(
          fit: BoxFit.contain, // Para alterar o tamanho do logo.
          child: const FlutterLogo(),
        ),
      ),
      Expanded(
        child: Text(
          "Olá, Mundo! Este é o meu primeiro aplicativo usando o widget Row. O  
widget Row dispõe seus filhos em linha.",
          textDirection: TextDirection.ltr,
          style: TextStyle(
```

```
        color: Colors.green,  
        fontSize: 30,  
        fontWeight: FontWeight.bold,  
        backgroundColor: Colors.white,  
      ), //TextStyle.  
    ),  
  ],  
));  
}
```



17) Altere o código novamente, como indicado abaixo.

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(  
    Row(  
      textDirection: TextDirection.ltr,  
      children: [  
        Expanded(  
          child: FittedBox(  
            fit: BoxFit.contain,  
            child: const FlutterLogo(),  
          ),  
        ],  
      ),  
    ),  
  ),  
);
```

```
Expanded(  
  child: FittedBox(  
    fit: BoxFit.contain,  
    child: Image(  
      image: NetworkImage(  
        'https://picsum.photos/250?image=9',  
      ),  
    ),  
  ),  
),  
Expanded(  
  child: Text(  
    "Olá, Mundo! Este é o meu primeiro aplicativo usando o widget Row.  
O widget Row dispõe seus filhos em linha.",  
    textDirection: TextDirection.ltr,  
    textAlign: TextAlign.center,  
    style: TextStyle(  
      color: Colors.green,  
      fontSize: 10,  
      fontWeight: FontWeight.bold,  
      backgroundColor: Colors.white,  
    ), //TextStyle.  
  ),  
),  
],  
),  
);  
}
```




Exercício

- 1) Altere o exemplo desta prática, para que ele apresente a tela abaixo.

Links para o exercício:

<https://picsum.photos/250?image=10>

<https://picsum.photos/250?image=25>

<https://picsum.photos/250?image=15>



Centro Universitário UNA
Flutter
Wesley Dias Maciel
2021/02

