



Flutter

Wesley Dias Maciel

2021/02

Prática 05

Column

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

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

O widget Column exibe seus filhos em um vetor vertical. Para que um filho do widget Column se expanda e preencha todo o espaço vertical disponível, envolva-o em um widget Expanded. O widget Column não possui a funcionalidade de rolagem (scroll). Em geral, é considerado um erro que o widget Column possua um ou mais filhos que não caibam no espaço disponível. Se você tem uma coluna de widgets e quer que eles possam rolar, caso não haja espaço suficiente na coluna, então, use um widget ListView. Para dispor widgets na direção horizontal, use o widget Row. Caso você tenha 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:

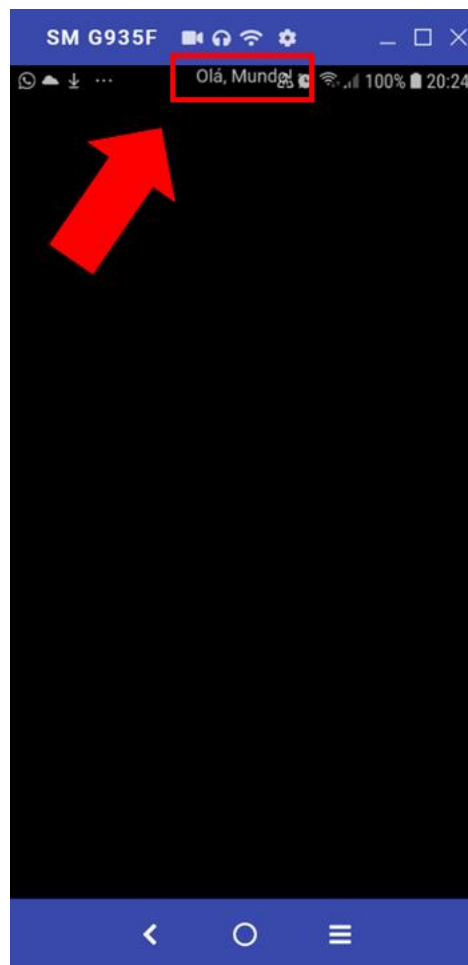
>Flutter: New Project
- 2) Na nova janela aberta, selecione um diretório para seu projeto. **OBS:** o caminho do diretório deve ter apenas caracteres ASCII. Além disso, o caminho do diretório não pode ter espaços em branco. Dessa forma, não use: acentos, símbolos, espaços, cedilha, etc. Exemplo: C:\projeto\flutter.
- 3) Na nova caixa de entrada, informe um nome para seu projeto. Exemplo: pratica05. 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. Após a criação de seu projeto, observe a estrutura de diretórios criada e o código inicial gerado como exemplo em lib\main.dart. Apague o código inicial gerado como exemplo em lib\main.dart.
- 5) Um aplicativo Flutter mínimo possui uma função principal main() que simplesmente chama a função runApp() com um widget.
- 6) No exemplo abaixo, a função runApp() é chamada com um widget para coluna (Column). O aplicativo apresenta na tela um texto.

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

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

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

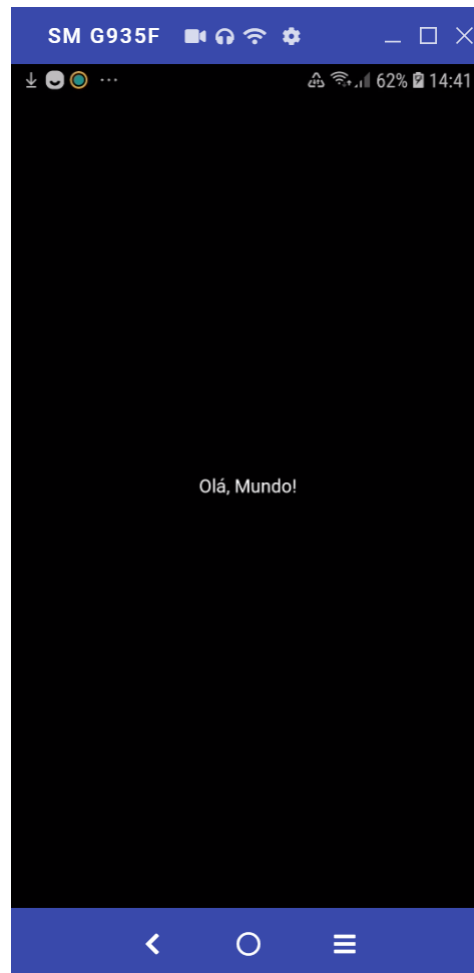


- 8) No widget Column, o eixo principal (**main axis**) é o eixo vertical, eixo y. Por outro lado, o eixo transversal (**cross axis**) é o eixo horizontal, eixo x. Dessa forma, os parâmetros `mainAxisAlignment` e `mainAxisSize` aplicam configuração de alinhamento e tamanho, respectivamente, ao eixo vertical,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

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

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

```
),  
],  
,  
);  
}
```

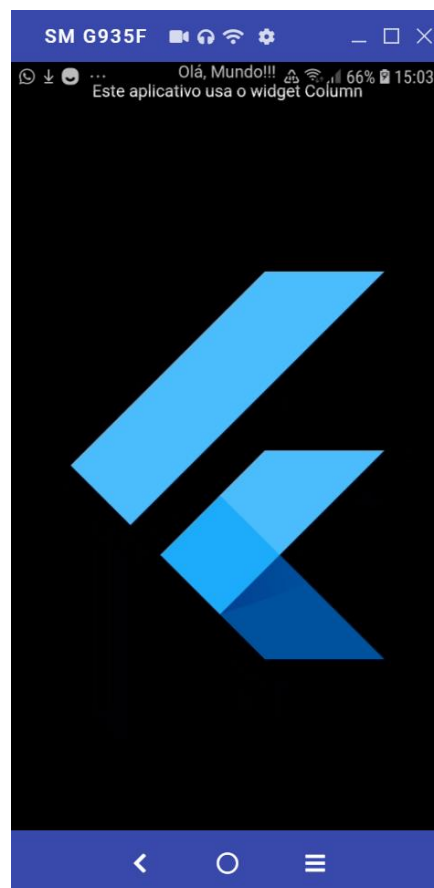


- 9) Use o widget Expanded para ampliar o espaço reservado para widgets filhos. O exemplo abaixo usa uma coluna para organizar três widgets verticalmente. O último widget é configurado para preencher todo o espaço restante na tela.

```
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(  
    Column(  
      children: [  
        Text(  

```

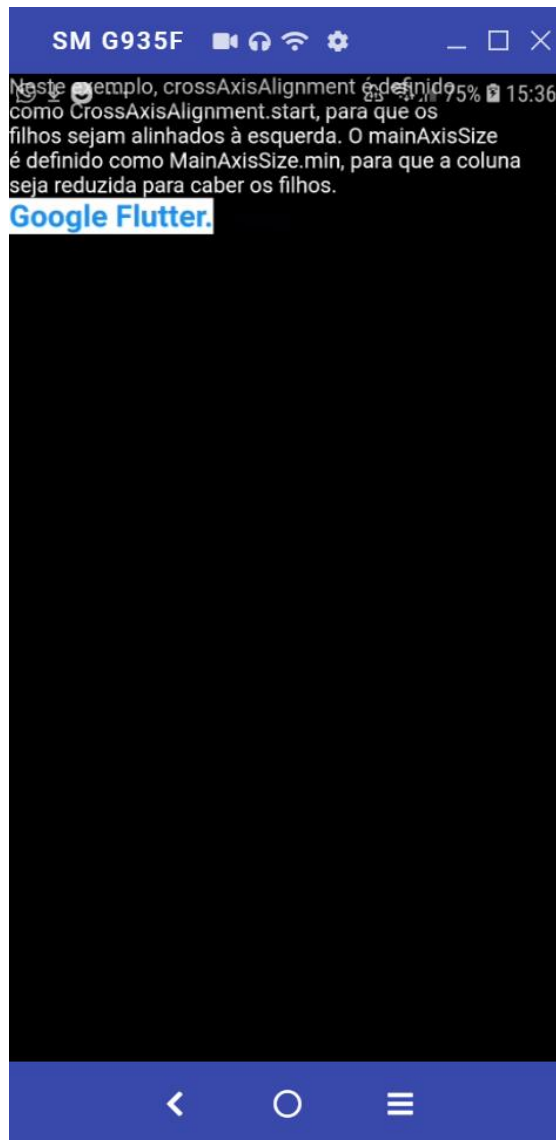
```
'Olá, Mundo!!!',  
  textDirection: TextDirection.ltr,  
),  
Text(  
  'Este aplicativo usa o widget Column',  
  textDirection: TextDirection.ltr,  
),  
Expanded(  
  child: FittedBox(  
    fit: BoxFit.contain, // Permite aumentar o tamanho do logo.  
    child: const FlutterLogo(),  
  ),  
),  
],  
),  
);  
}
```



- 10) No exemplo abaixo, `crossAxisAlignment` é definido como `CrossAxisAlignment.start`, para que os filhos sejam alinhados à esquerda. O `mainAxisSize` é definido como `MainAxisSize.min`, para que a coluna seja reduzida ao tamanho dos filhos.

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

void main() {
  runApp(
    Column(
      textDirection: TextDirection.ltr,
      crossAxisAlignment: CrossAxisAlignment.start,
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        Text(
          'Neste exemplo, crossAxisAlignment é definido',
          textDirection: TextDirection.ltr,
        ),
        Text(
          'como CrossAxisAlignment.start, para que os ',
          textDirection: TextDirection.ltr,
        ),
        Text(
          'filhos sejam alinhados à esquerda. O mainAxisAlignment ',
          textDirection: TextDirection.ltr,
        ),
        Text(
          'é definido como MainAxisAlignment.min, para que a coluna ',
          textDirection: TextDirection.ltr,
        ),
        Text(
          'seja reduzida para caber os filhos.',
          textDirection: TextDirection.ltr,
        ),
        Text(
          'Google Flutter.',
          textDirection: TextDirection.ltr,
          style: TextStyle(
            color: Colors.blue,
            fontSize: 20,
            fontWeight: FontWeight.bold,
            backgroundColor: Colors.white,
          ),
        ),
      ],
    ),
  );
}
```



Exercício

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

Link para o exercício: <https://picsum.photos/250?image=9>

