



Flutter

Wesley Dias Maciel

2021/02



Centro Universitário UNA
Flutter
Wesley Dias Maciel
2021/02

Flutter

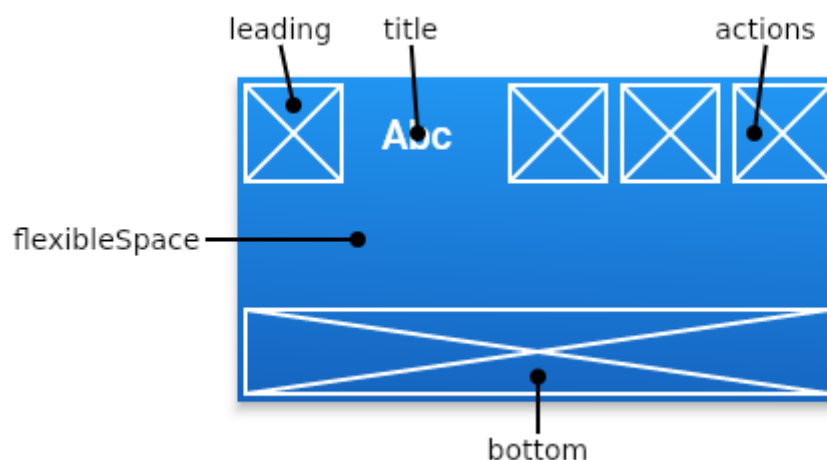
Prática 19

AppBar

Documentação: <https://api.flutter.dev/flutter/material/AppBar-class.html>

Objetivo: exemplificar o uso do widget AppBar.

O widget AppBar é uma barra de ferramentas que contém outros widgets. Um AppBar geralmente expõe uma ou mais ações através de ícones. Além disso, são normalmente usados na propriedade Scaffold.appBar, que coloca o AppBar como um widget de altura fixa na parte superior da tela. Um AppBar exibe os widgets: leading, title e actions. O diagrama a seguir mostra onde esses widgets aparecem:



Nesta prática, você vai:

- a) Criar um AppBar.
 - b) Inserir widgets nesse AppBar.
 - c) Associar funcionalidades aos widgets do AppBar.
- 1) Crie um novo projeto Flutter, usando:
- a. Visual Studio Code, ou;
 - b. Visual Studio Code, ou;
 - c. <https://dartpad.dev/>, ou;
 - d. <https://flutlab.io/ide>, ou;

- e. <https://flutterstudio.app/>, ou;
- f. <https://codemagic.io/>.

AppBar

- 2) O exemplo abaixo cria duas rotas. A primeira rota possui um widget AppBar. O AppBar possui:
- a. leading: que recebe um widget IconButton.
 - b. title: que recebe um widget Text.

Ao clicar no ícone da primeira rota, o usuário é direcionado para a segunda rota.

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

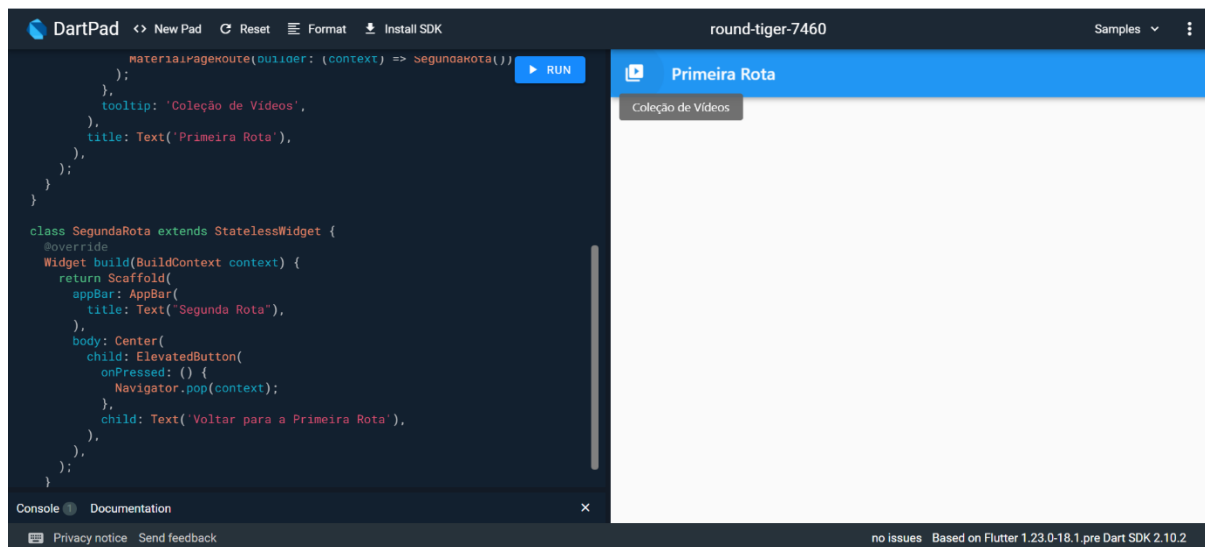
void main() => runApp(MeuAplicativo());

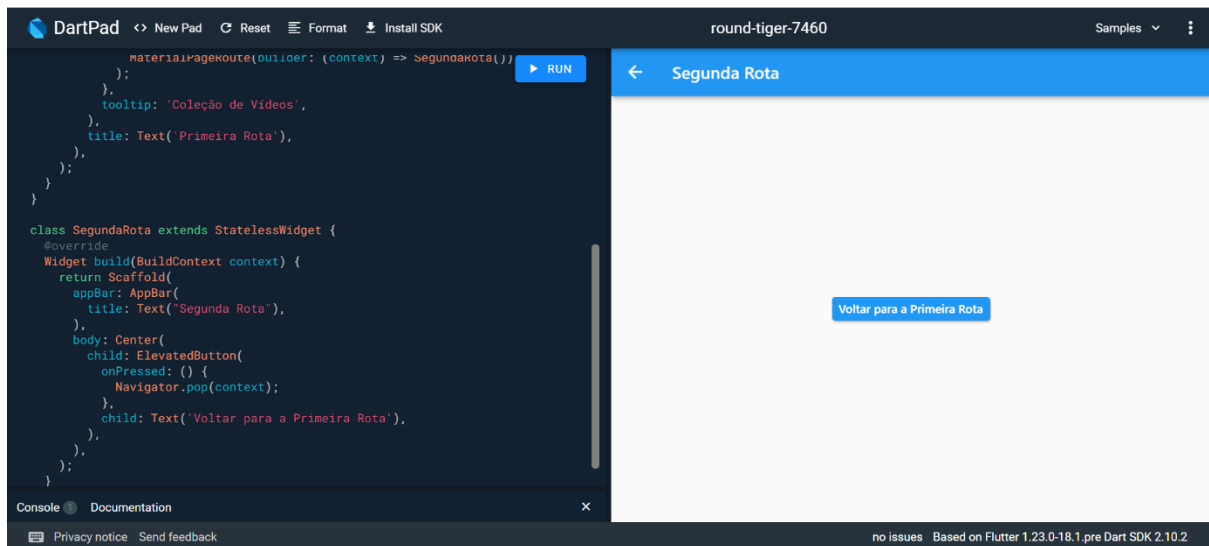
class MeuAplicativo extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: PrimeiraRota(),
      debugShowCheckedModeBanner: false,
    );
  }
}

class PrimeiraRota extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        leading: IconButton(
          icon: Icon(Icons.video_collection),
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (context) => SegundaRota()),
            );
          },
        ),
        tooltip: 'Coleção de Vídeos',
      ),
      title: Text('Primeira Rota'),
    );
  }
}
```



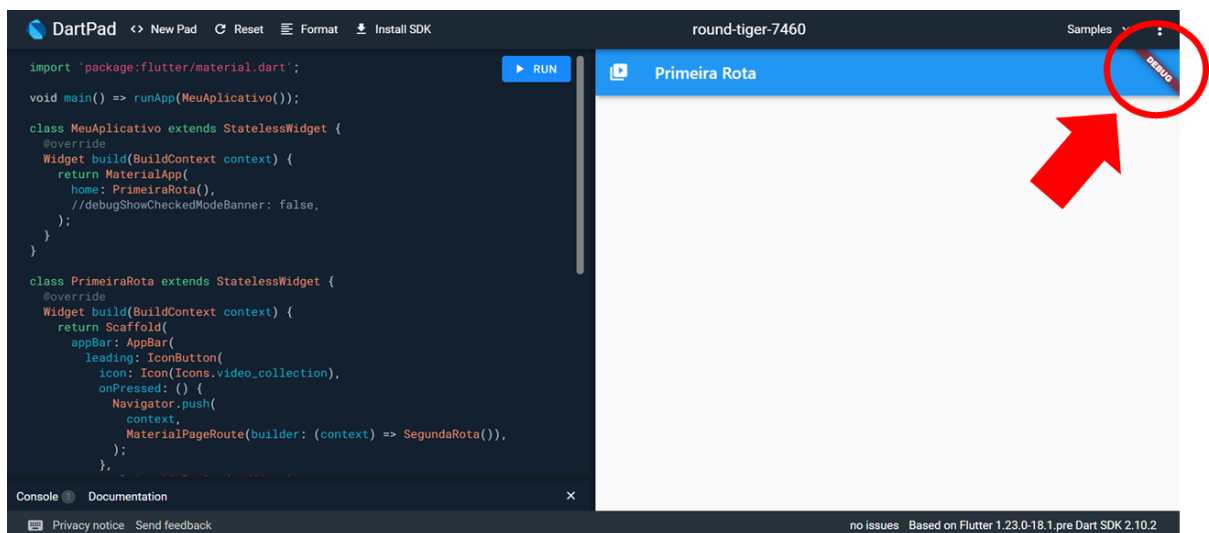
```
class SegundaRota extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Center(  
        child: ElevatedButton(  
          onPressed: () {  
            Navigator.pop(context);  
          },  
          child: Text('Voltar para a Primeira Rota'),  
        ),  
      ),  
    );  
  }  
}
```





OBS: o atributo `debugShowCheckedModeBanner` do widget `MaterialApp` recebeu o valor falso, para ocultar a indicação do modo de depuração, debug:

`debugShowCheckedModeBanner: false,`



actions

- 3) O widget `actions` permite inserir uma lista de widgets no `AppBar`. No exemplo abaixo, o widget `actions` possui uma lista de ícones. Ao clicar num dos dois primeiros ícones, o algoritmo apresenta um cartão com o meio de transporte relacionado ao ícone. Ao clicar

no terceiro ícone, o algoritmo apresenta um menu. Ao clicar numa opção do menu, o algoritmo também apresenta um cartão com o meio de transporte correspondente.

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

class Transporte {
  final String titulo;
  final IconData icone;

  const Transporte(this.titulo, this.icone);
}

const List<Transporte> transportes = <Transporte>[
  Transporte('Carro', Icons.directions_car),
  Transporte('Bicicleta', Icons.directions_bike),
  Transporte('Barco', Icons.directions_boat),
  Transporte('Ônibus', Icons.directions_bus),
  Transporte('Trem', Icons.directions_railway),
];

void main() => runApp(MeuAplicativo());

class MeuAplicativo extends StatelessWidget {
  @override
  Widget build(BuildContext context) => MaterialApp(
    home: PrimeiraRota(),
    debugShowCheckedModeBanner: false,
  );
}

class PrimeiraRota extends StatefulWidget {
  @override
  PrimeiraRotaState createState() => PrimeiraRotaState();
}

class PrimeiraRotaState extends State<PrimeiraRota> {
  Transporte transporte = transportes[0];

  void selecionar(Transporte transporteEscolhido) =>
    setState(() => this.transporte = transporteEscolhido);

  List<PopupMenuItem<Transporte>> getListaItensMenu() {
    List<PopupMenuItem<Transporte>> lista;

    lista = transportes.skip(2).map(
      (Transporte transporte) {
```

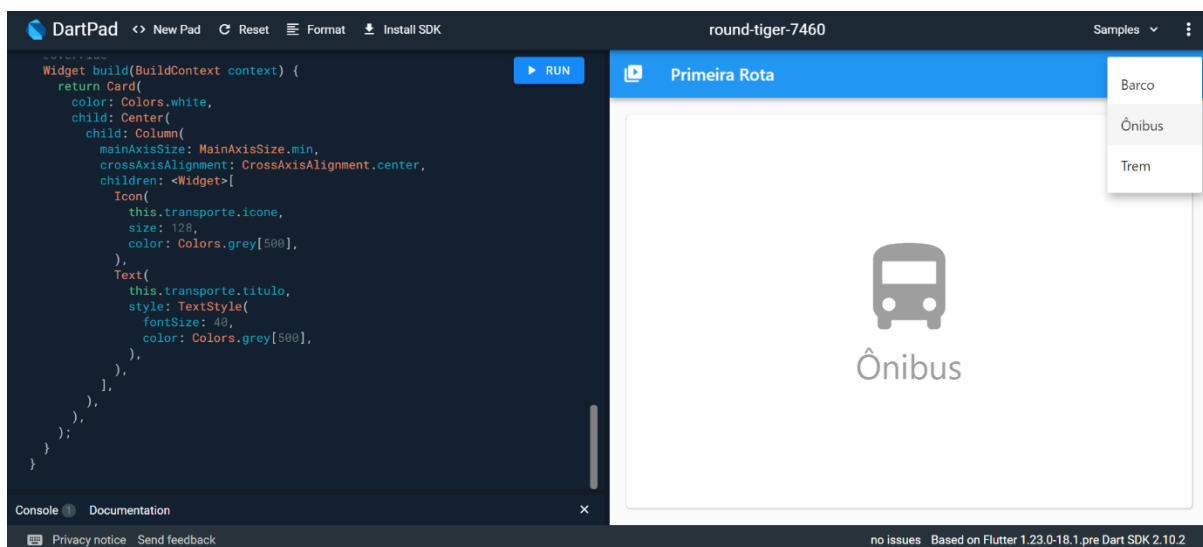
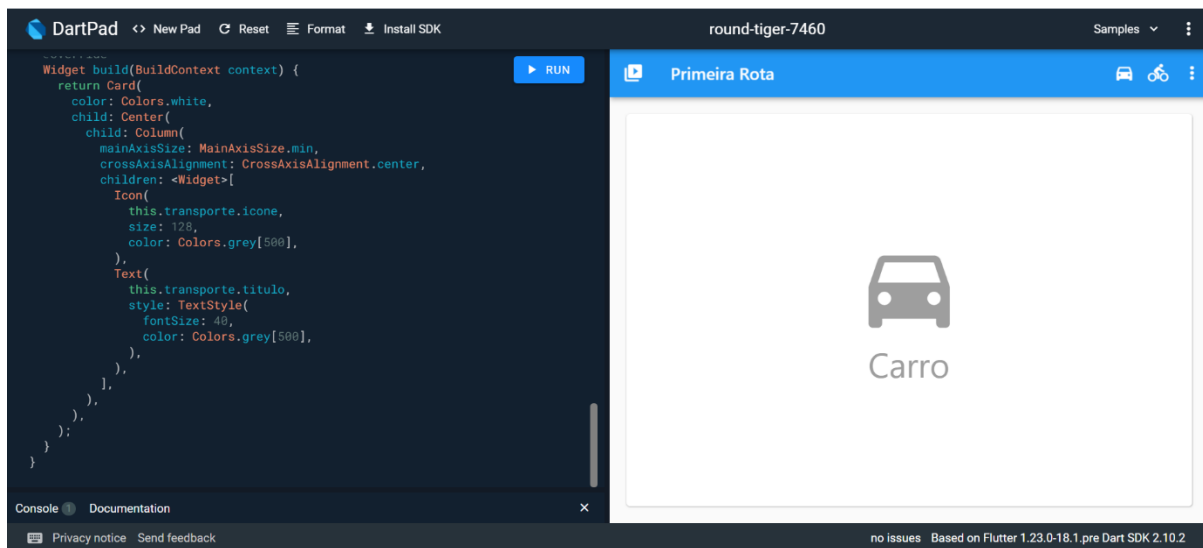
```
        return PopupMenuItem<Transporte>(  
            value: transporte,  
            child: Text(transporte.titulo),  
        );  
    },  
).toList();  
  
return lista;  
}  
  
@override  
Widget build(BuildContext context) {  
    return Scaffold(  
        appBar: AppBar(  
            leading: IconButton(  
                icon: Icon(Icons.video_collection),  
                onPressed: () => Navigator.push(  
                    context,  
                    MaterialPageRoute(builder: (context) => SegundaRota()),  
                ),  
            tooltip: 'Coleção de Vídeos',  
        ),  
        title: Text('Primeira Rota'),  
        actions: <Widget>[  
            IconButton(  
                icon: Icon(transportes[0].icone),  
                onPressed: () => selecionar(transportes[0]),  
            ),  
            IconButton(  
                icon: Icon(transportes[1].icone),  
                onPressed: () => selecionar(transportes[1]),  
            ),  
            PopupMenuButton<Transporte>(  
                onSelected: selecionar,  
                itemBuilder: (BuildContext context) => getListaItensMenu(),  
            ),  
        ],  
    ),  
    body: Padding(  
        padding: EdgeInsets.all(16),  
        child: Cartao(this.transporte),  
    ),  
);  
}  
}  
  
class SegundaRota extends StatelessWidget {
```



```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("Segunda Rota"),
    ),
    body: Center(
      child: ElevatedButton(
        onPressed: () => Navigator.pop(context),
        child: Text('Voltar para a Primeira Rota'),
      ),
    ),
  );
}

class Cartao extends StatelessWidget {
  final Transporte transporte;
  Cartao(this.transporte);

  @override
  Widget build(BuildContext context) {
    return Card(
      color: Colors.white,
      child: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.min,
          crossAxisAlignment: CrossAxisAlignment.center,
          children: <Widget>[
            Icon(
              this.transporte.icone,
              size: 128,
              color: Colors.grey[500],
            ),
            Text(
              this.transporte.titulo,
              style: TextStyle(
                fontSize: 40,
                color: Colors.grey[500],
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```



.map()

Na linguagem Dart, as listas possuem o método `.map()`. O método `.map()` recebe uma função como parâmetro e aplica esta função a cada elemento da lista. O retorno do método `.map()` é um `Iterable`. Entretanto, caso você queira que o retorno seja uma lista, use `.map().toList()`. Exemplo:

```
//map() -> retorna um Iterable
//map().toList() -> retorna uma lista.

void main() {
```

```
List nums = [5, 6, 7, 8, 9];

print('Lista:');
print(nums); //Lista.

print('Iterable:');
print(nums.map((n) => 2 * n)); //Iterable.
print('Lista:');
print(nums); //Lista.

print('Lista:');
print(nums.map((n) => 2 * n).toList()); //Lista.
print('Lista:');
print(nums); //Lista.
}
```

Exercício

- 1) Altere o algoritmo apresentado nesta prática, para que ao clicar num item do menu, o usuário seja direcionado para uma nova rota. A nova rota deve apresentar o cartão correspondente à opção selecionada no menu. A nova rota deve ser gerada por uma rota genérica.

Exemplo:

