



# Centro Universitário UNA

Usabilidade, Desenvolvimento Web,  
Mobile e Jogos

Graduação – TI e Engenharias

Práticas de Laboratório

Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria  
Barros, Wesley Dias Maciel

2020/02



Centro Universitário UNA  
Graduação – TI e Engenharias  
Usabilidade, Desenvolvimento Web, Mobile e Jogos  
Prática de Laboratório  
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto  
de Faria Barros, Wesley Dias Maciel  
2020/02

# Flutter



## Prática 15

# Retornar Valor de uma Rota para Outra

Documentação: <https://flutter.dev/docs/cookbook/navigation/returning-data>,  
<https://api.flutter.dev/flutter/widgets/Navigator-class.html>

**Objetivo:** considerando um aplicativo com duas rotas, retornar um valor processado na segunda rota para a primeira através do método `Navigator.then ()`.

Nesta prática, você vai:

- 1) Criar as duas rotas.
- 2) Navegar da primeira para a segunda rota usando o método `Navigator.push ()`.
  - Usar o método `Navigator.then ()` para atualizar a primeira rota com a função `setState ()`.
- 3) Usar o método `Navigator.pop ()` para voltar para a primeira rota com o valor gerado na segunda.

- 1) Crie um novo projeto Flutter, usando:
  - a. Visual Studio Code, ou;
  - b. <https://dartpad.dev/>, ou;
  - c. <https://flutlab.io/>, ou;
  - d. <https://flutterstudio.app/>, ou;
  - e. <https://codemagic.io/>.

## Criar duas Rotas

- 2) O exemplo abaixo cria as duas rotas. Cada rota é um widget.

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

void main() => runApp(
  MaterialApp(
    home: PrimeiraRota(),
  ),
);
```



```
);

class PrimeiraRota extends StatefulWidget {
  PrimeiraRotaState createState() {
    return PrimeiraRotaState();
  }
}

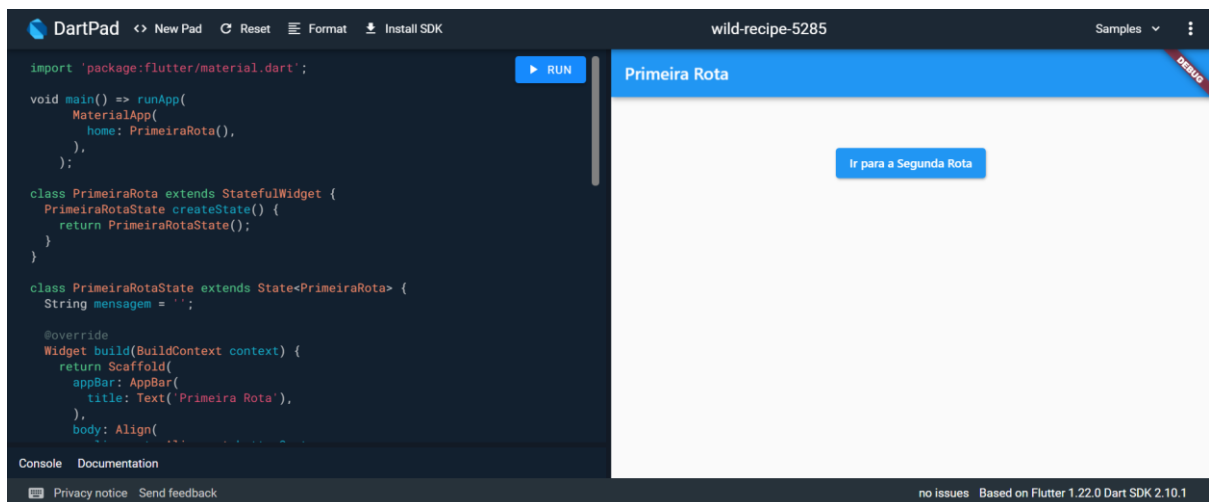
class PrimeiraRotaState extends State<PrimeiraRota> {
  String mensagem = '';

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Primeira Rota'),
      ),
      body: Center(
        child: Column(
          children: [
            Text(
              '$mensagem',
              style: TextStyle(
                fontSize: 40,
                color: Colors.green,
                fontWeight: FontWeight.bold,
              ),
            ),
            ElevatedButton(
              child: Text('Ir para a Segunda Rota'),
              onPressed: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(builder: (context) => SegundaRota()),
                ).then((resposta) {
                  setState(() {
                    mensagem = resposta;
                  });
                });
              },
            ),
          ],
        ),
      ),
    );
  }
}
```



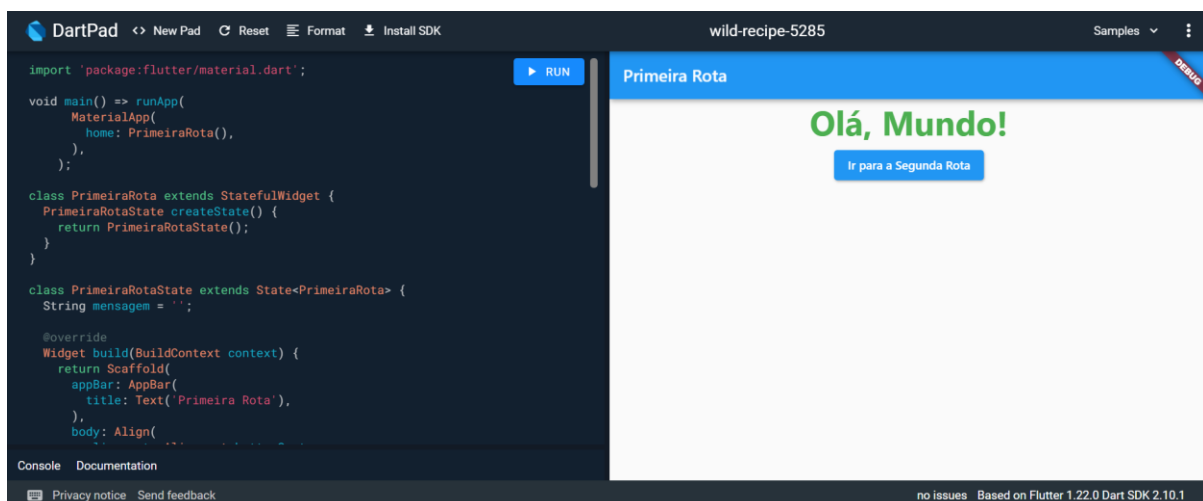
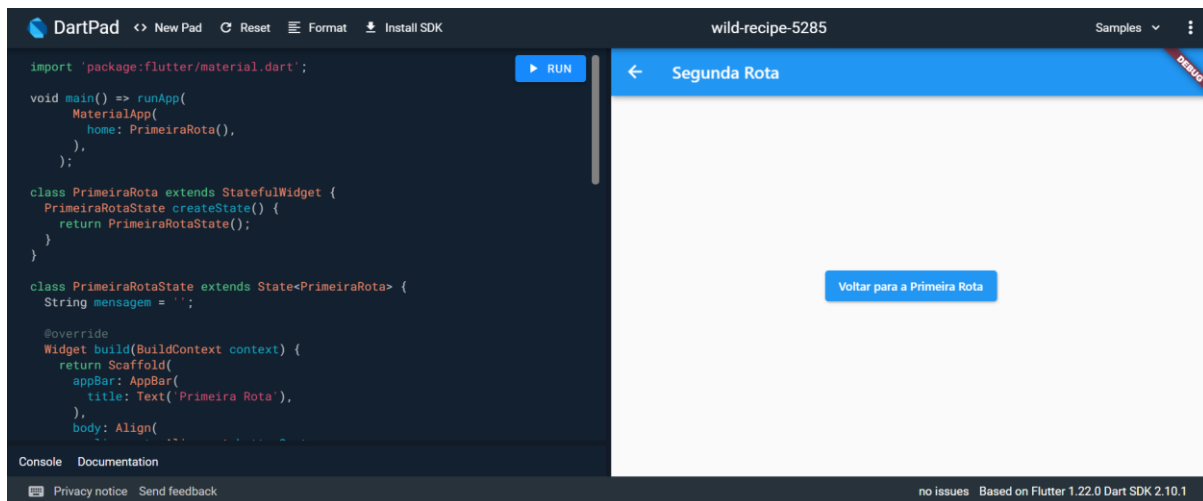
Centro Universitário UNA  
Graduação – TI e Engenharias  
Usabilidade, Desenvolvimento Web, Mobile e Jogos  
Prática de Laboratório  
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria Barros, Wesley Dias Maciel  
2020/02

```
);  
}  
}  
  
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, 'Olá, Mundo!');  
          },  
          child: Text('Voltar para a Primeira Rota'),  
        ),  
      ),  
    );  
  }  
}
```





Centro Universitário UNA  
Graduação – TI e Engenharias  
Usabilidade, Desenvolvimento Web, Mobile e Jogos  
Prática de Laboratório  
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria Barros, Wesley Dias Maciel  
2020/02



## Navigator.push () e Navigator.then ()

- 3) No atributo onPressed do botão da primeira rota, permitir a navegação através do método Navigator.push (). Além disso, usar o método Navigator.then () para atualizar a primeira rota com a função setState ().

O método Navigator.then () só será executado depois que a segunda rota retornar uma resposta através de Navigator.pop (), nesse exemplo a resposta é do tipo String. Essa resposta é atribuída ao objeto mensagem, que é usado na atualização da primeira rota.



```
onPressed: () {  
  Navigator.push(  
    context,  
    MaterialPageRoute(builder: (context) => SegundaRota()),  
  ).then((resposta) {  
    setState(() {  
      mensagem = resposta;  
    });  
  });  
},
```

## Navigator.pop ()

- 4) No atributo onPressed do botão da segunda rota, usar o método Navigator.pop () para voltar para a primeira rota, retornando o valor gerado. Nesse exemplo, o valor retornado é o String 'Olá, Mundo!'.

```
onPressed: () {  
  Navigator.pop(context, 'Olá, Mundo!');  
},
```

## Exercício

- 1) Altere o algoritmo apresentado nesta prática, de forma que:
  - a. A primeira rota tenha um botão que direcione o usuário para a segunda rota.
  - b. A segunda rota permita que o usuário informe dois números. Ao clicar no botão da segunda rota, o usuário é direcionado para a primeira rota juntamente com a soma dos dois números informados.
  - c. Ao retornar, a primeira rota deve apresentar o resultado da soma dos números informados.



Centro Universitário UNA  
Graduação – TI e Engenharias  
Usabilidade, Desenvolvimento Web, Mobile e Jogos  
Prática de Laboratório  
Carlos Augusto dos Santos Pinheiro, Cristiano de Macedo Neto, Diego Augusto de Faria Barros, Wesley Dias Maciel  
2020/02

DartPad wild-recipe-5285

```
class SegundaRota extends StatelessWidget {  
  TextEditingController num01Controller = TextEditingController();  
  TextEditingController num02Controller = TextEditingController();  
  String resp = "";  
  
  somar() {  
    int num01 = int.tryParse(this.num01Controller.text);  
    int num02 = int.tryParse(this.num02Controller.text);  
    int soma = num01 + num02;  
    this.resp = '$num01 + $num02 = $soma';  
    return this.resp;  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Column(  

```

Primeira Rota

Ir para a Segunda Rota

1 issues show Based on Flutter 1.22.0 Dart SDK 2.10.1

DartPad wild-recipe-5285

```
class SegundaRota extends StatelessWidget {  
  TextEditingController num01Controller = TextEditingController();  
  TextEditingController num02Controller = TextEditingController();  
  String resp = "";  
  
  somar() {  
    int num01 = int.tryParse(this.num01Controller.text);  
    int num02 = int.tryParse(this.num02Controller.text);  
    int soma = num01 + num02;  
    this.resp = '$num01 + $num02 = $soma';  
    return this.resp;  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Column(  

```

Segunda Rota

informe o primeiro número  
2

informe o segundo número  
3

Voltar para a Primeira Rota

1 issues show Based on Flutter 1.22.0 Dart SDK 2.10.1

DartPad wild-recipe-5285

```
class SegundaRota extends StatelessWidget {  
  TextEditingController num01Controller = TextEditingController();  
  TextEditingController num02Controller = TextEditingController();  
  String resp = "";  
  
  somar() {  
    int num01 = int.tryParse(this.num01Controller.text);  
    int num02 = int.tryParse(this.num02Controller.text);  
    int soma = num01 + num02;  
    this.resp = '$num01 + $num02 = $soma';  
    return this.resp;  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Rota"),  
      ),  
      body: Column(  

```

Primeira Rota

2 + 3 = 5

Ir para a Segunda Rota

1 issues show Based on Flutter 1.22.0 Dart SDK 2.10.1