

Flutter

Aula 07

Prof. Dr. Rodrigo Plotze

rodrigoplotze@gmail.com



- ElevatedButton
- OutlinedButton
- TextButton
- Navigator

ELEVATED BUTTON

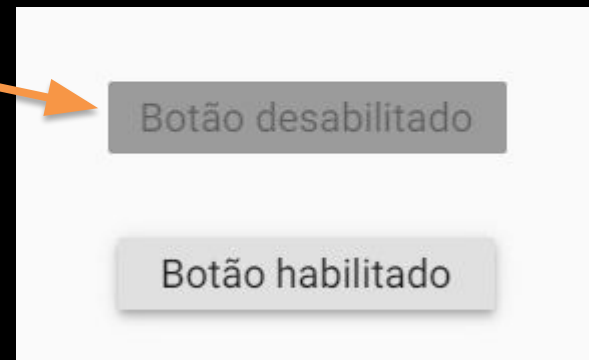




- Widget baseado no Material que apresenta uma “elevação” quando o botão é pressionado.
- Eventos
 - `onPressed`
 - `onLongPress`
- Se os retornos de chamada *onPressed* e *onLongPress* forem nulos, o botão será desativado.

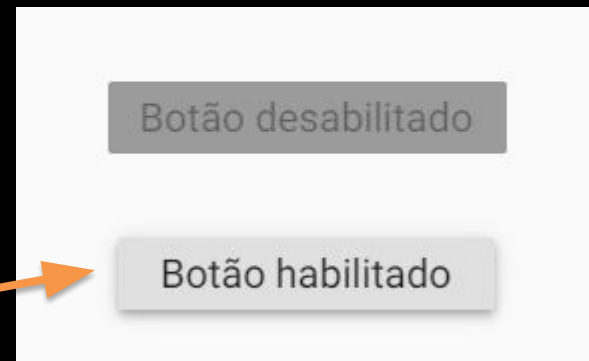


```
RaisedButton(  
  onPressed: null,  
  child: Text('Botão desabilitado'),  
)  
  
SizeBox(height: 30),  
  
RaisedButton(  
  onPressed: () {},  
  onLongPress: () {},  
  child: const Text(' Botão habilitado '),  
)
```





```
RaisedButton(  
  onPressed: null,  
  child: Text('Botão desabilitado'),  
)  
  
SizeBox(height: 30),  
  
RaisedButton(  
  onPressed: () {},  
  onLongPress: () {},  
  child: const Text(' Botão habilitado '),  
)
```





Exemplo de utilização do ElevatedButton com ícone:

```
1 ElevatedButton.icon(  
2   label: Text('Woolha.com'),  
3   icon: Icon(Icons.web),  
4   onPressed: () {  
5     print('Pressed');  
6   },  
7 )
```

Output:





Especificação da aparência do botão com tema:

```
1  import 'package:flutter/material.dart';
2
3  void main() => runApp(MyApp());
4
5  class MyApp extends StatelessWidget {
6
7    @override
8    Widget build(BuildContext context) {
9      return MaterialApp(
10        title: 'Woolha.com Flutter Tutorial',
11        home: _ElevatedButtonExample(),
12        theme: ThemeData(
13          elevatedButtonTheme: ElevatedButtonThemeData(
14            style: ElevatedButton.styleFrom(
15              primary: Colors.purple,
16            ),
17          ),
18        ),
19      );
20    }
21  }
22
23  class _ElevatedButtonExample extends StatelessWidget {
24
25    @override
26    Widget build(BuildContext context) {
27      return Scaffold(
28        appBar: AppBar(
29          title: Text('Woolha.com Flutter Tutorial'),
30        ),
31        body: Center(
32          child: ElevatedButton(
33            child: Text('Woolha.com'),
34            onPressed: () {
35              print('Pressed');
36            },
37          ),
38        ),
39      );
40    }
41  }
42
```




Especificação da aparência do botão:

```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5   ),  
6   onPressed: () {  
7     print('Pressed');  
8   },  
9 )
```

Output:



```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5     onPrimary: Colors.white,  
6     onSurface: Colors.grey,  
7   ),  
8   onPressed: () {  
9     print('Pressed');  
10  },  
11 )
```

Output:





Especificação do estilo do texto do botão:

```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5     onPrimary: Colors.white,  
6     textStyle: TextStyle(  
7       color: Colors.black,  
8       fontSize: 40,  
9       fontStyle: FontStyle.italic  
10    ),  
11  ),  
12  onPressed: () {  
13    print('Pressed');  
14  },  
15 )
```

Output:



Woolha.com



Especificação do sombreamento e da elevação:

```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5     onPrimary: Colors.white,  
6     shadowColor: Colors.red,  
7     elevation: 5,  
8   ),  
9   onPressed: () {  
10     print('Pressed');  
11   },  
12 )
```

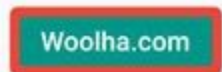




Especificação da borda:

```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5     onPrimary: Colors.white,  
6     side: BorderSide(color: Colors.red, width: 5),  
7   ),  
8   onPressed: () {  
9     print('Pressed');  
10  },  
11 )
```

Output:





Especificação do formato:

```
1 ElevatedButton(  
2   child: Text('Woolha.com'),  
3   style: ElevatedButton.styleFrom(  
4     primary: Colors.teal,  
5     onPrimary: Colors.white,  
6     shape: const BeveledRectangleBorder(borderRadius: BorderRadius.all(Radius.circular(5))),  
7   ),  
8   onPressed: () {  
9     print('Pressed');  
10  },  
11 )
```

Output:



Woolha.com

OUTLINED BUTTON





- Este widget permite a inserção de botões com borda de maneira fácil e prática.

```
1 OutlinedButton(  
2   child: Text('Woolha.com'),  
3   onPressed: () {  
4     print('Pressed');  
5   },  
6 )
```

Output:



Woolha.com

TEXT BUTTON





- Este botão tem como principal característica a aparência “flat”, sem borda e sem elevação:

```
1 TextButton(  
2   child: Text('Woolha.com'),  
3   onPressed: () {  
4     print('Pressed');  
5   }  
6 )
```

Output:



```
1 TextButton.icon(  
2   label: Text('Woolha.com'),  
3   icon: Icon(Icons.web),  
4   onPressed: () {  
5     print('Pressed');  
6   }  
7 )
```

Output:



NAVIGATOR






- A maioria dos aplicativos contém várias telas para exibir diferentes tipos de informações.
 - Por exemplo, um aplicativo pode ter uma tela que exibe produtos.
 - Quando o usuário toca na imagem de um produto, uma nova tela exibe detalhes sobre o produto.
- A navegação entre ***rotas*** é realizada pela classe ***Navigator***.

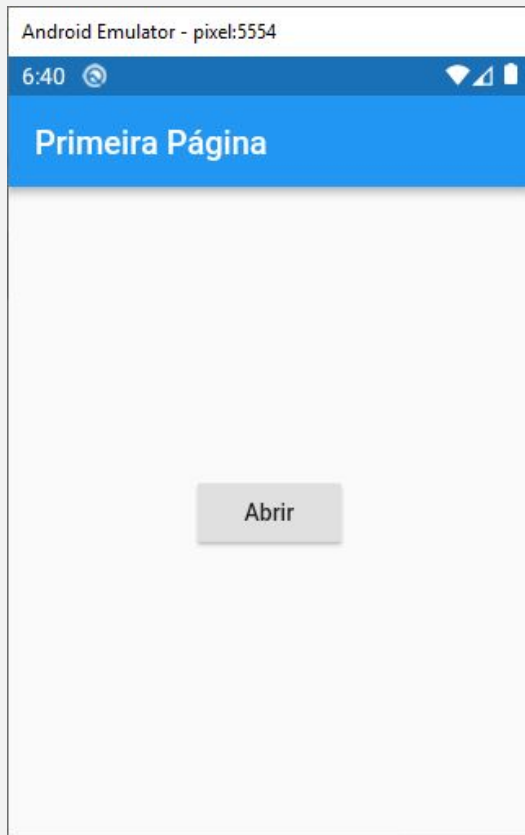



```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    home: PrimeiraTela(),  
  ));  
}
```



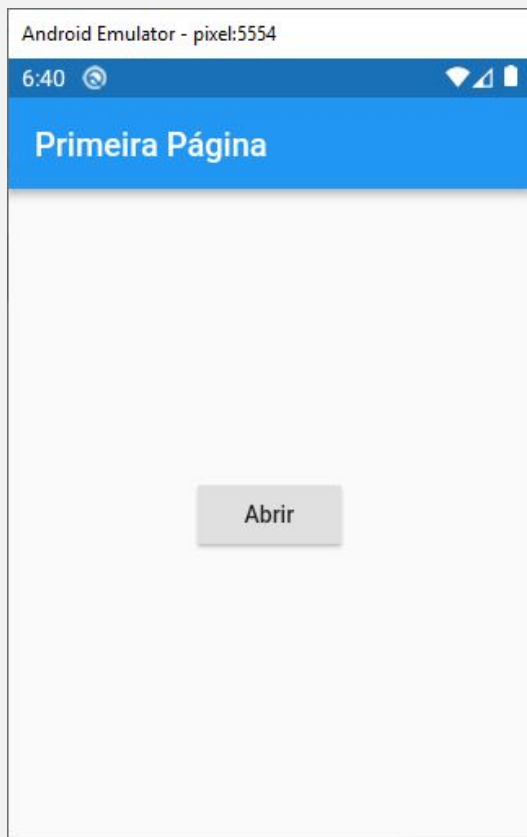



```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    home: PrimeiraTela(),  
  ));  
}
```





```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    home: PrimeiraTela(),  
  ));  
}
```

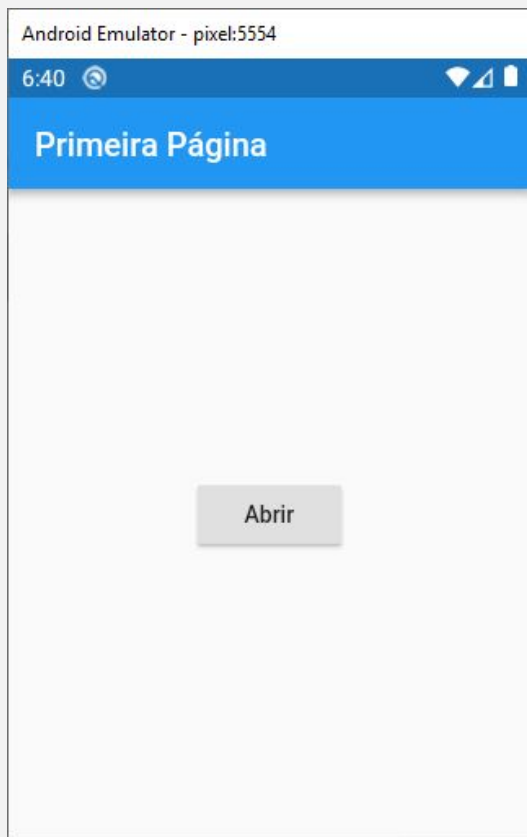



`Navigator.push`





```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    home: PrimeiraTela(),  
  ));  
}
```



`Navigator.push`

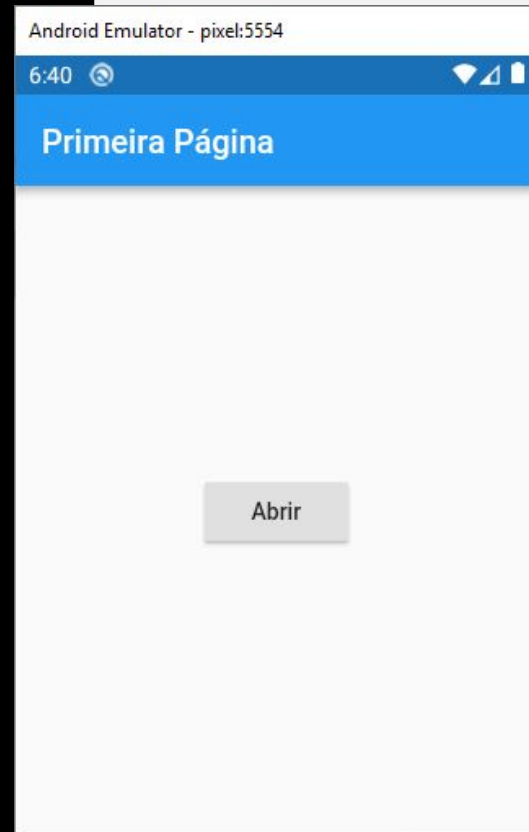


`Navigator.pop`



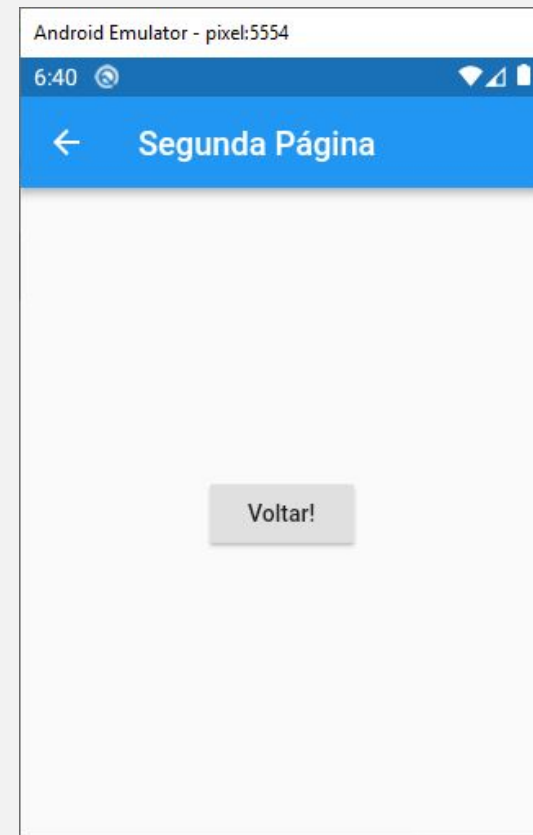


```
class PrimeiraTela extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Primeira Página'),  
      ),  
      body: Center(  
        child: RaisedButton(  
          child: Text('Abrir'),  
          onPressed: () {  
            Navigator.push(context,  
              MaterialPageRoute(builder:  
                (context) => SegundaTela()  
            ),  
          ),  
        ),  
      ),  
    );  
  }  
}
```





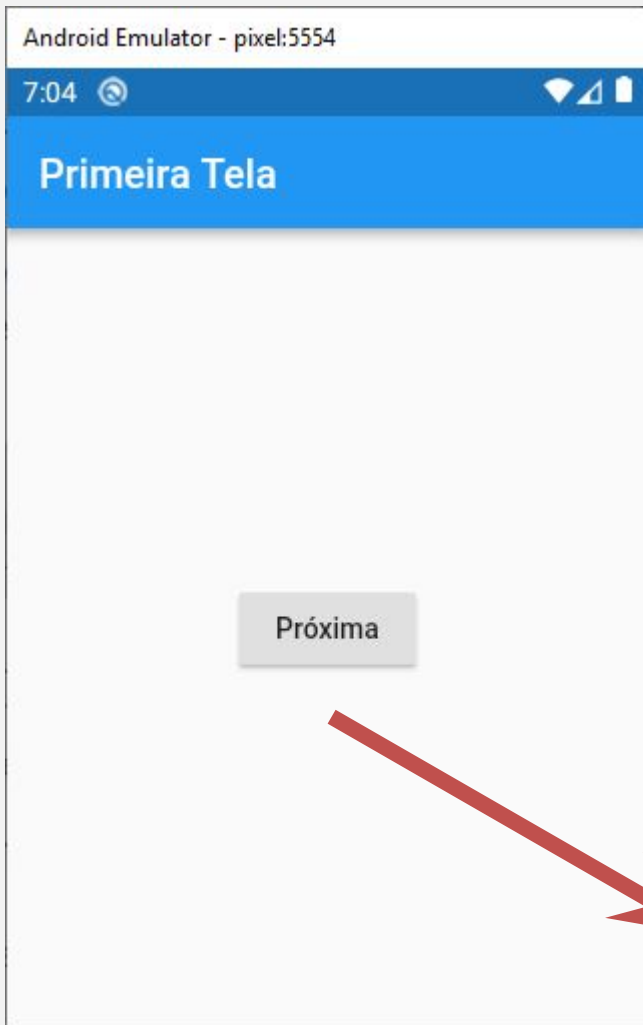
```
class SegundaTela extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Segunda Página"),  
      ),  
      body: Center(  
        child: RaisedButton(  
          onPressed: () {  
            Navigator.pop(context);  
          },  
          child: Text('Voltar!'),  
        ),  
      ),  
    );  
  }  
}
```



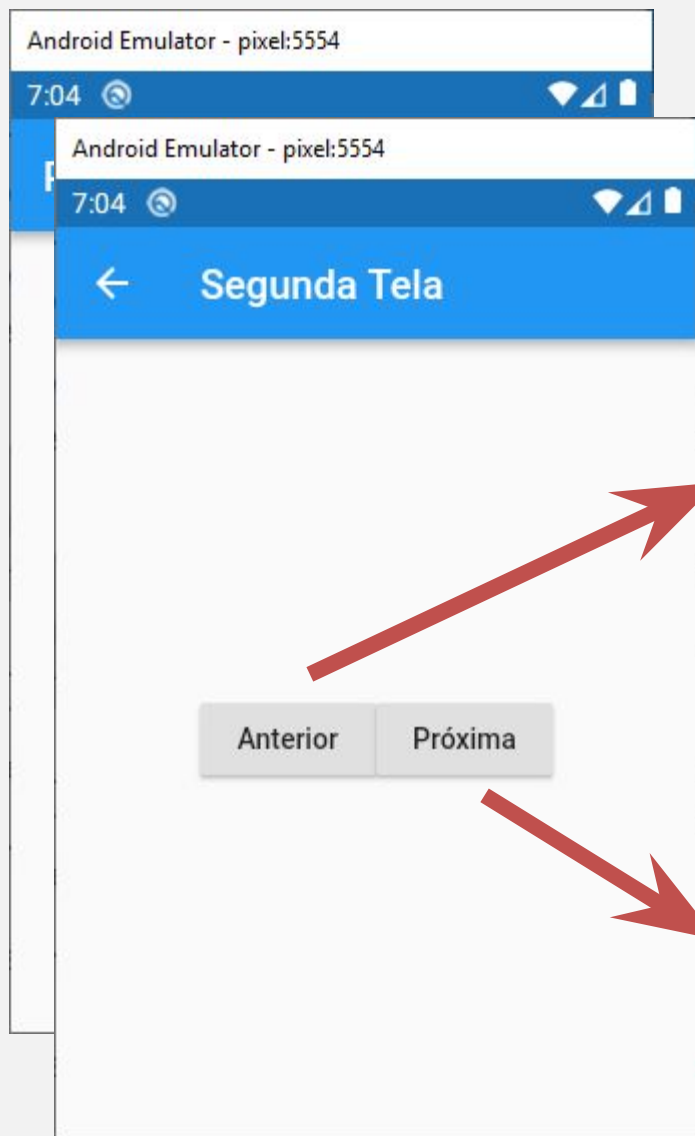


- Uma alternativa para navegação entre as rotas do aplicativo é o uso de *rotas nomeadas*.

```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    initialRoute: '/tela1',  
    routes: {  
      '/tela1': (context) => PrimeiraTela(),  
      '/tela2': (context) => SegundaTela(),  
      '/tela3': (context) => TerceiraTela(),  
      '/tela4': (context) => QuartaTela(),  
    },  
  ));  
}
```

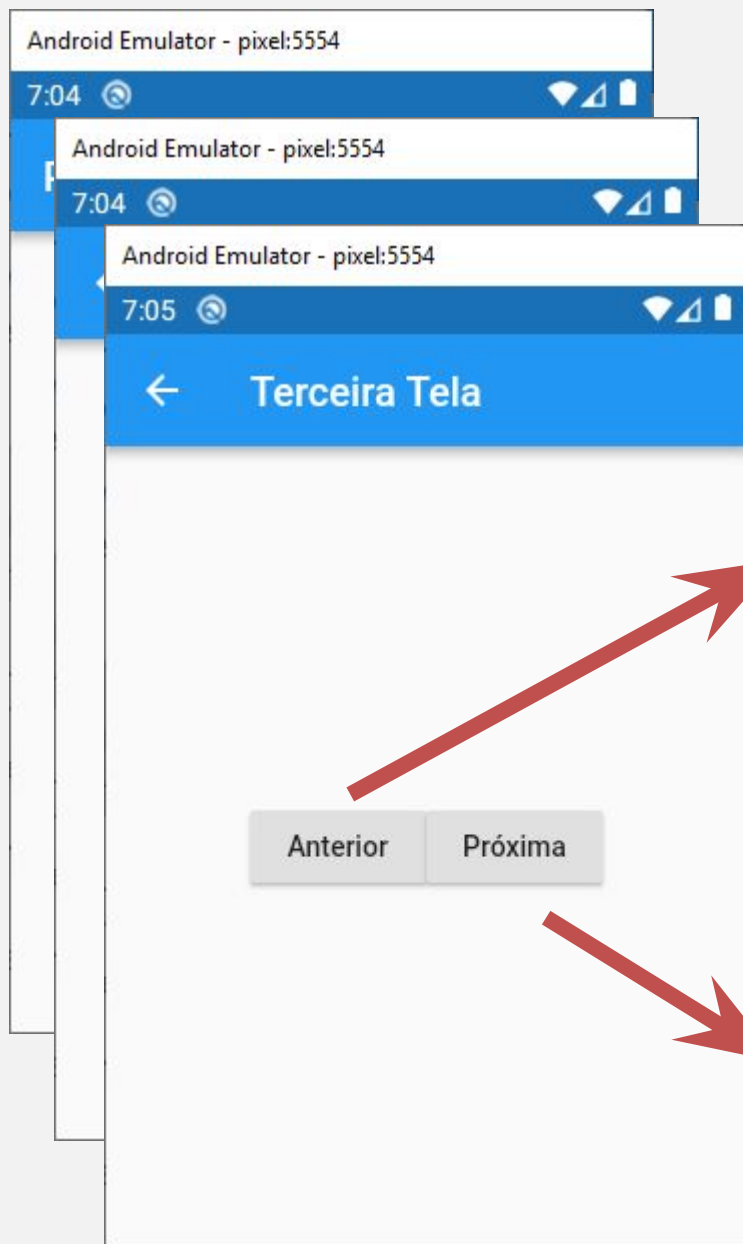


```
Navigator.pushNamed(context, '/tela2');
```



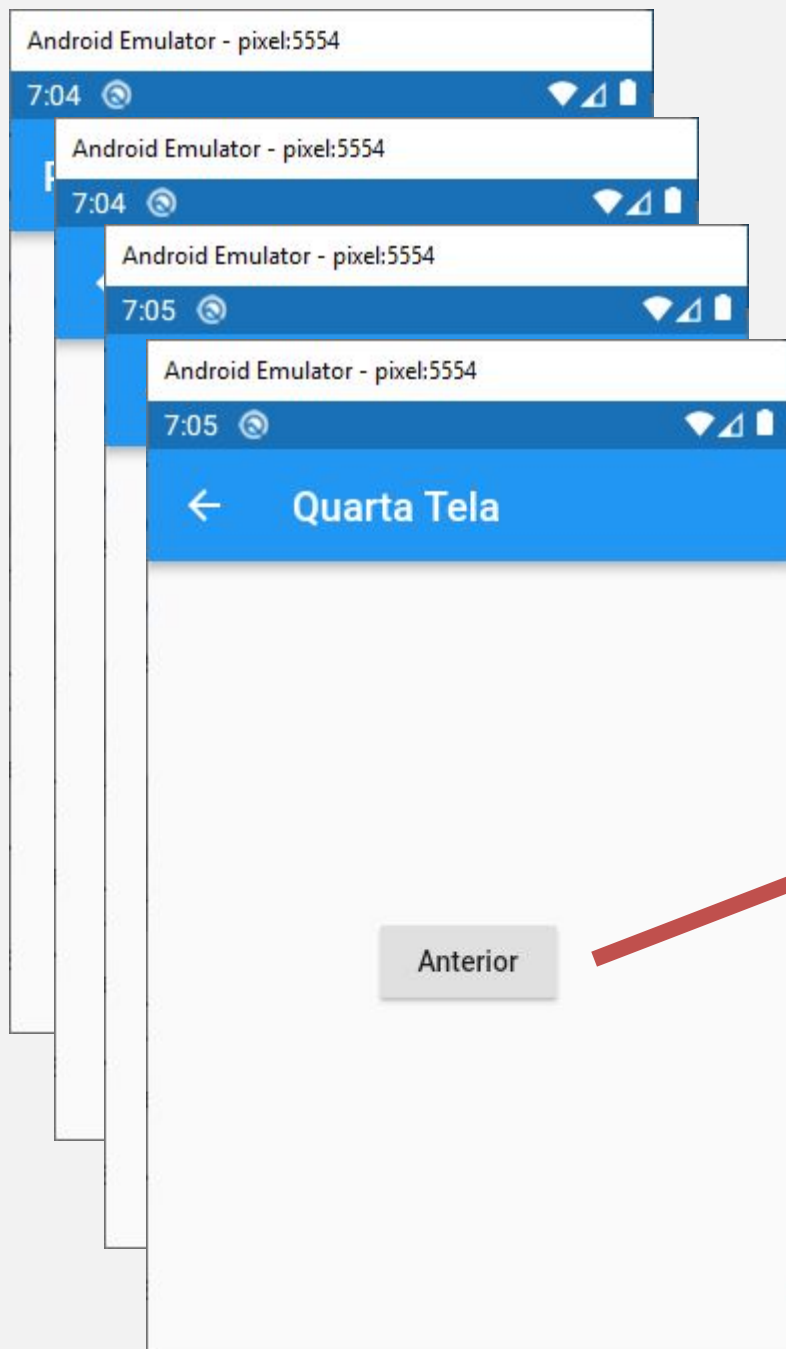
```
Navigator.pop(context);
```

```
Navigator.pushNamed(context, '/tela3');
```



```
Navigator.pop(context);
```

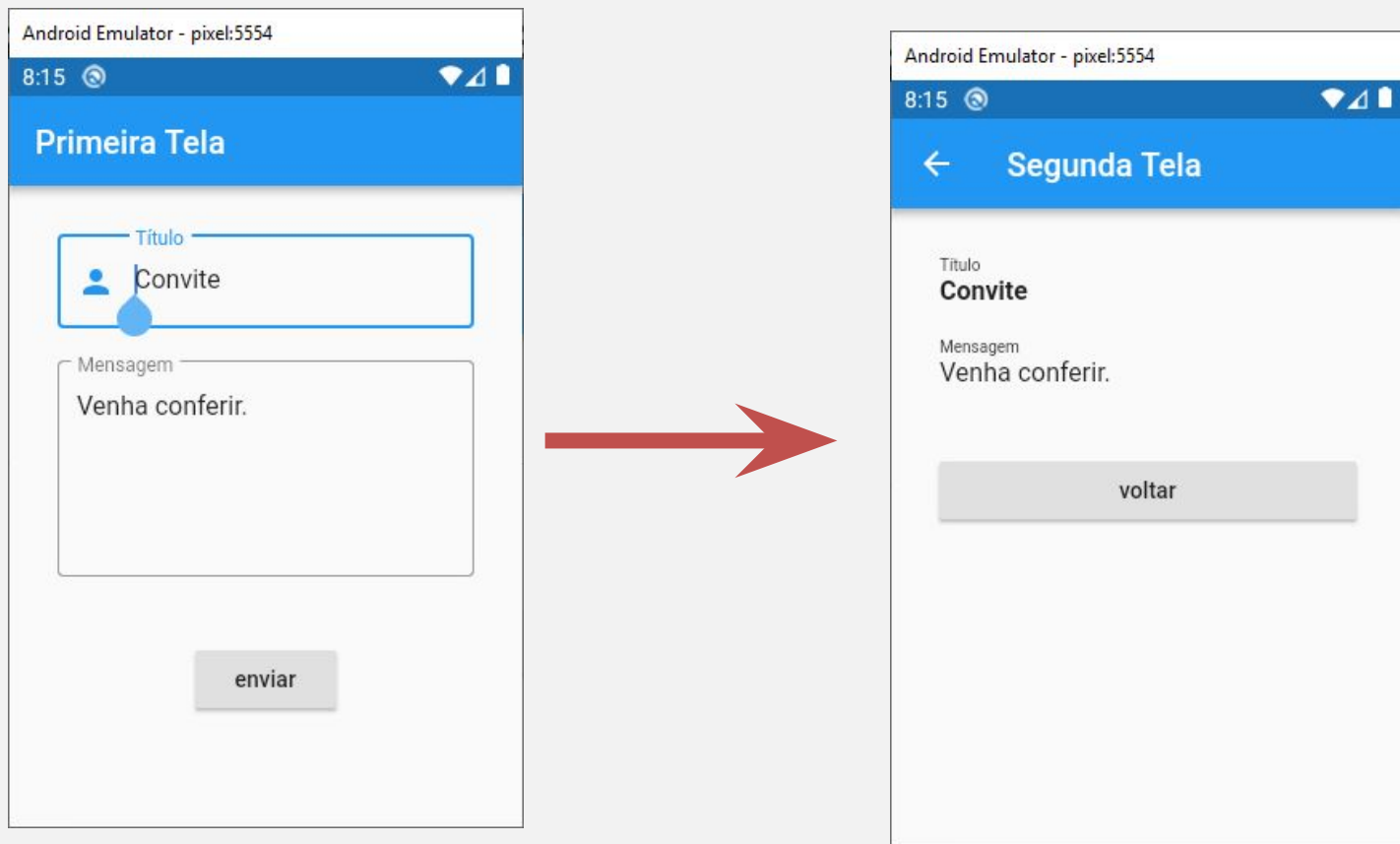
```
Navigator.pushNamed(context, '/tela4');
```



Anterior

```
Navigator.pop(context);
```

- O ***navigator*** permite passar argumentos para uma rota nomeada.



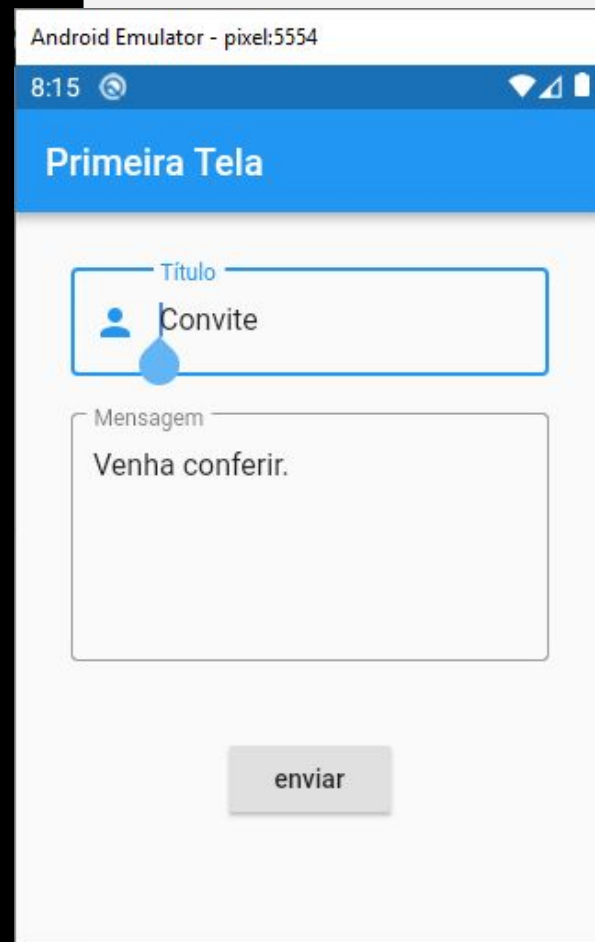


```
void main() {  
  runApp(MaterialApp(  
    debugShowCheckedModeBanner: false,  
    title: 'Navegação',  
    initialRoute: '/tela1',  
    routes: {  
      '/tela1': (context) => PrimeiraTela(),  
      '/tela2': (context) => SegundaTela(),  
    },  
  ));  
}
```

```
class Mensagem {  
  final String titulo;  
  final String mensagem;  
  Mensagem(this.titulo, this.mensagem);  
}
```



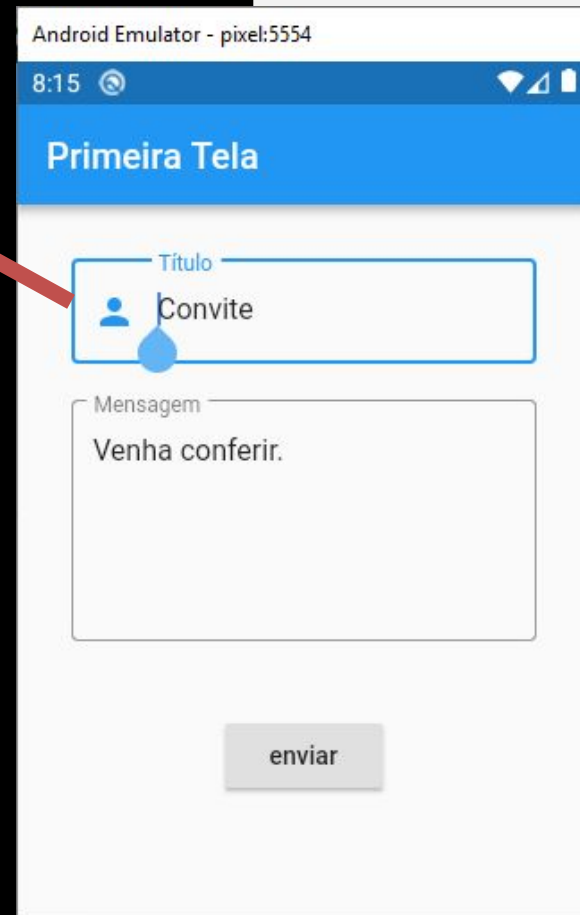

```
class PrimeiraTela extends StatefulWidget {  
  @override  
  _PrimeiraTelaState createState() => _PrimeiraTelaState();  
}  
  
class _PrimeiraTelaState extends State<PrimeiraTela> {  
  
  var _titulo = "";  
  var _mensagem = "";  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text('Primeira Tela'),  
      ),  
      body: SingleChildScrollView(  
        padding: EdgeInsets.all(30),  
        child: Column(  
          children: <Widget>[  
  
            /* INSERIR WIDGETS da 1ª TELA */  
  
          ],  
        ),  
      ),  
    );  
  }  
}
```





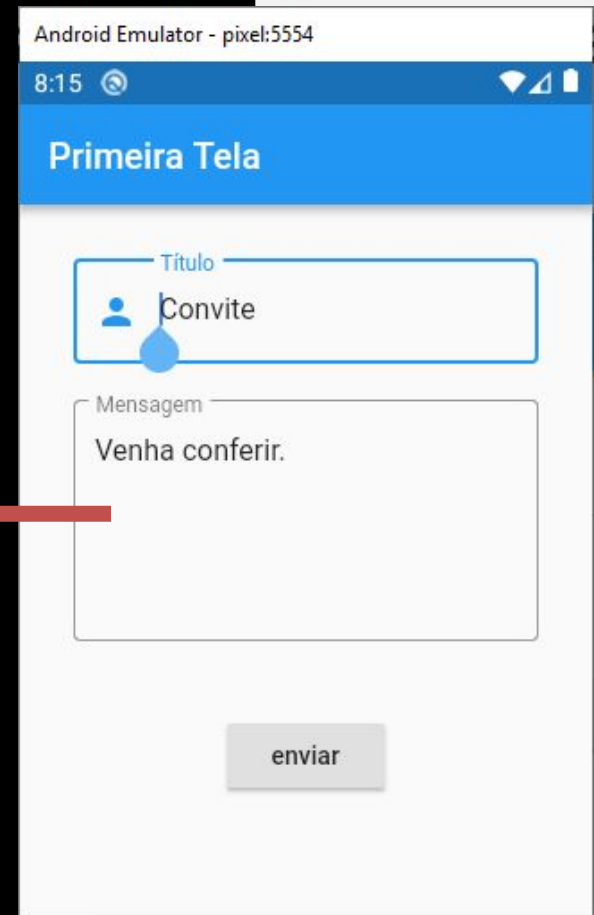
```
TextField(  
  decoration: InputDecoration(  
    prefixIcon: Icon(Icons.person),  
    border: OutlineInputBorder(),  
    labelText: 'Título',  
  ),  
  onChanged: (text) { _titulo = text; },  
,  
  SizedBox(height: 20),  
  TextField(  
    keyboardType: TextInputType.multiline,  
    maxLines: 5,  
    decoration: InputDecoration(  
      border: OutlineInputBorder(),  
      labelText: 'Mensagem', ),  
    onChanged: (text) { _mensagem = text; },  
  ),  
  SizedBox(height: 40),  
  RaisedButton(  
    child: Text('enviar'),  
    onPressed: () {  
      Navigator.pushNamed(context, SegundaTela.nomeRota,  
        arguments: Mensagem(_titulo, _mensagem));  
    },  
  ),  
)
```

Widgets da 1ª TELA



Widgets da 1ª TELA

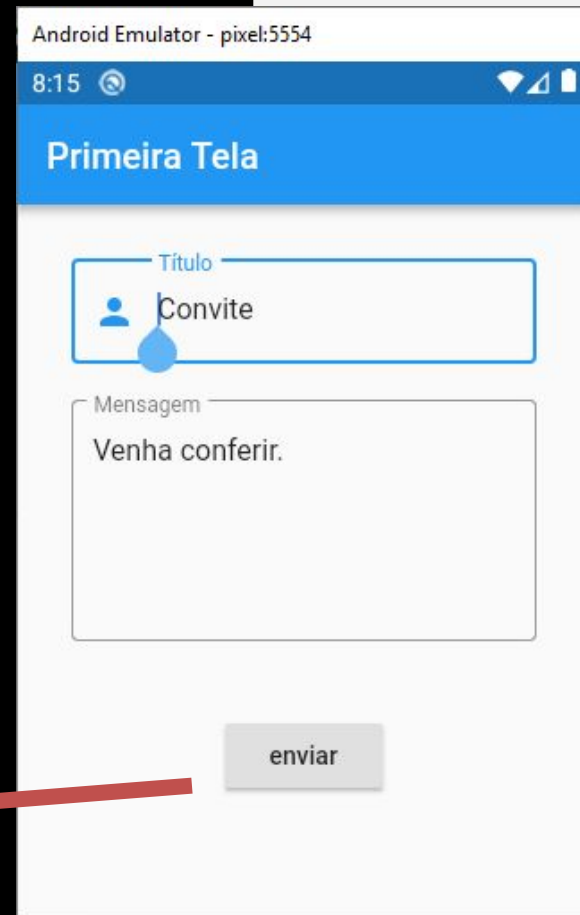
```
TextField(  
  decoration: InputDecoration(  
    prefixIcon: Icon(Icons.person),  
    border: OutlineInputBorder(),  
    labelText: 'Título',  
  ),  
  onChanged: (text) { _titulo = text; },  
)  
,  
  SizedBox(height: 20),  
  TextField(  
    keyboardType: TextInputType.multiline,  
    maxLines: 5,  
    decoration: InputDecoration(  
      border: OutlineInputBorder(),  
      labelText: 'Mensagem', ),  
    onChanged: (text) { _mensagem = text; },  
  ),  
  SizedBox(height: 40),  
  RaisedButton(  
    child: Text('enviar'),  
    onPressed: () {  
      Navigator.pushNamed(context, SegundaTela.nomeRota,  
        arguments: Mensagem(_titulo, _mensagem));  
    },  
  ),  
)
```





```
TextField(  
  decoration: InputDecoration(  
    prefixIcon: Icon(Icons.person),  
    border: OutlineInputBorder(),  
    labelText: 'Título',  
  ),  
  onChanged: (text) { _titulo = text; },  
,  
  SizedBox(height: 20),  
  TextField(  
    keyboardType: TextInputType.multiline,  
    maxLines: 5,  
    decoration: InputDecoration(  
      border: OutlineInputBorder(),  
      labelText: 'Mensagem', ),  
    onChanged: (text) { _mensagem = text; },  
  ),  
  SizedBox(height: 40),  
  RaisedButton(  
    child: Text('enviar'),  
    onPressed: () {  
      Navigator.pushNamed(context, SegundaTela.nomeRota,  
        arguments: Mensagem(_titulo, _mensagem));  
    },  
  ),  
)
```

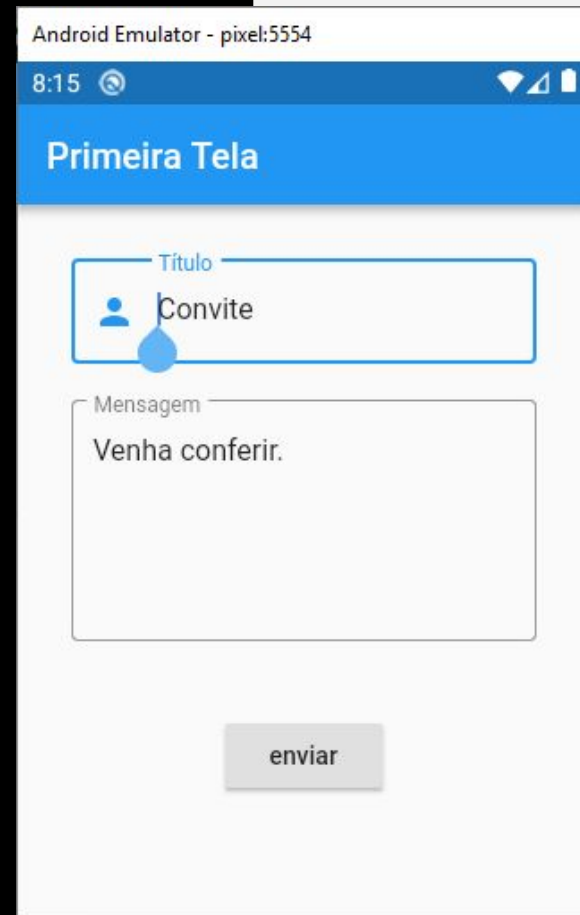
Widgets da 1ª TELA





```
TextField(  
  decoration: InputDecoration(  
    prefixIcon: Icon(Icons.person),  
    border: OutlineInputBorder(),  
    labelText: 'Título',  
  ),  
  onChanged: (text) { _titulo = text; },  
,  
  SizedBox(height: 20),  
  TextField(  
    keyboardType: TextInputType.multiline,  
    maxLines: 5,  
    decoration: InputDecoration(  
      border: OutlineInputBorder(),  
      labelText: 'Mensagem', ),  
    onChanged: (text) { _mensagem = text; },  
  ),  
  SizedBox(height: 40),  
  RaisedButton(  
    child: Text('enviar'),  
    onPressed: () {  
      Navigator.pushNamed(context, SegundaTela.nomeRota,  
        arguments: Mensagem(_titulo, _mensagem));  
    },  
  ),  
,
```

Widgets da 1ª TELA



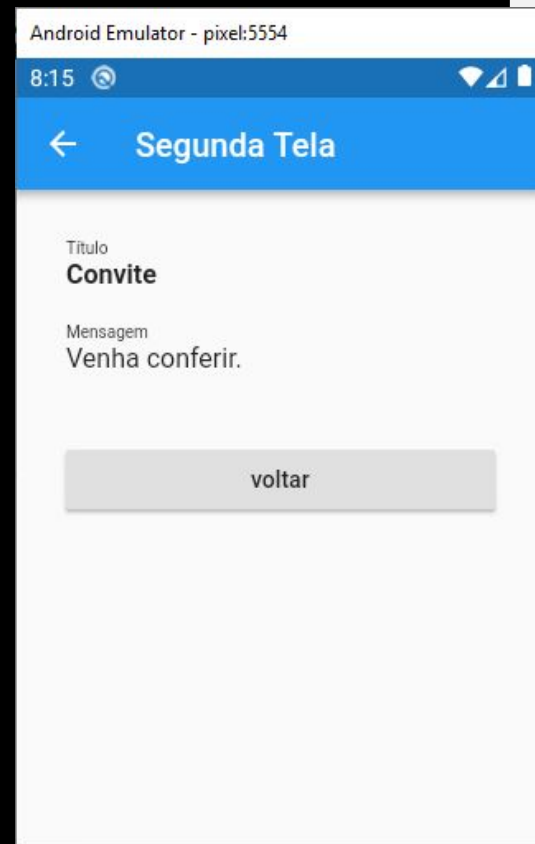
Passagem de parâmetros
entre as telas



```
class SegundaTela extends StatelessWidget {
  static const nomeRota = '/tela2';

  @override
  Widget build(BuildContext context) {
    final Mensagem msg = ModalRoute.of(context).settings.arguments;

    return Scaffold(
      appBar: AppBar(title: Text("Segunda Tela"),),
      body: SingleChildScrollView(padding: EdgeInsets.all(30),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: <Widget>[
            Text("Título", style: TextStyle(fontSize: 10)),
            Text(msg.titulo, style: TextStyle(fontSize: 16,
              fontWeight: FontWeight.bold)),
            SizedBox(height: 20,),
            Text("Mensagem", style: TextStyle(fontSize: 10)),
            Text(msg.mensagem, style: TextStyle(fontSize: 16)),
            SizedBox(height: 40, ),
            RaisedButton(child: Text('voltar'),
              onPressed: () {Navigator.pop(context); },),
          ],
        ),
      ),
    );
  }
}
```






Android Emulator - pixel:5554

8:15

Primeira Tela

Título

 Convite

Mensagem

Venha conferir.

enviar

Android Emulator - pixel:5554

8:15

← Segunda Tela

Título

Convite

Mensagem

Venha conferir.

voltar

FIM