Programação para Dispositivos Móveis


Introdução ao Flutter

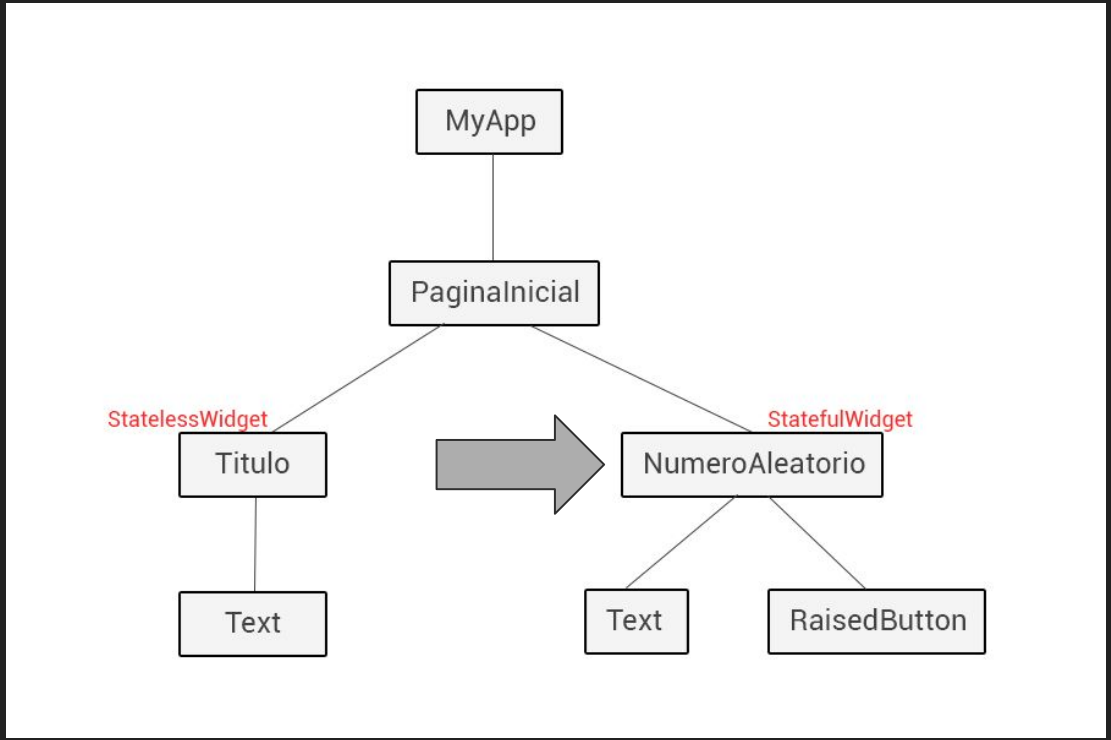
Introdução

Iremos nos basear no exemplo de app ao lado para explicar os fundamentos do Flutter.

Temos um app com os seguintes componentes (Widgets):

- Um label estático de título;
- Um botão que, quando clicado, irá gerar um número aleatório;
- Um label para exibir o número aleatório gerado.

Estrutura do exemplo



RUN

```
      appBar: AppBar(
        title: Text("Gerador aleatório")
      ),
      body: Center(
        child: Titulo()
      )
    );

  }

}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}
```


Gerador aleatório

Codificamos uma nova classe que represente o Widget de Geração de Números Aleatórios.

Ele será StatefulWidget já que seu estado será alterado em tempo de execução.

DartPad New Pad Reset Format Install SDK calm-wind-6600 Samples

RUN

```
        child: Titulo()
      )
    );

  }

}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

}
```

Console 1 Documentation

Gerador aleatório

Toda classe que seja Stateful dependerá de instâncias de uma classe de estado, onde de fato criamos os Widgets dinâmicos, definimos eventos a serem disparados pelos mesmos e como os estados dos Widgets serão alterados.

Privacy notice Send feedback 3 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

RUN

```
}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

  }

}
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Primeiro, vamos criar a GUI associada com a apresentação dos números aleatórios.

RUN

```
}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

    return Container();

  }
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

O Widget principal será um Container.

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```dart
class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

    return Container(
      child: Column()
    );

  }
```

Console 1 Documentation

Gerador aleatório

Nele, aninhamos um Widget Column. A partir dele, será possível aninhar vários Widgets em uma coluna com várias linhas (um abaixo do outro).

Privacy notice Send feedback 3 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```dart
class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

    return Container(
      child: Column(
        children: []
      )
    );
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Para aninharmos vários Widgets, utilizamos a propriedade children (filhos), atribuindo-os a uma lista.

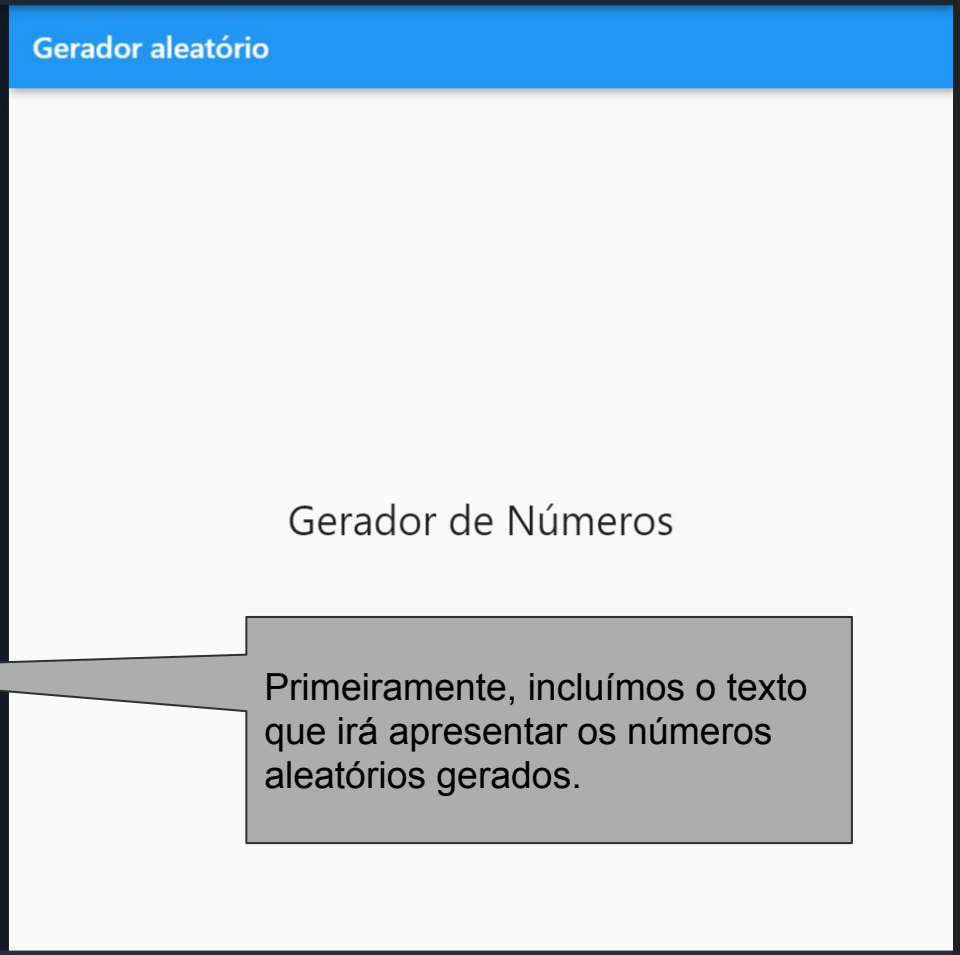
Privacy notice Send feedback 3 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```
    return Container(
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28))
        ]
      )
    );

  }

}
```

Console 1 Documentation



Privacy notice Send feedback 3 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context){

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30)
        ]
      )
    );

  }

}
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Logo em seguida, um Widget SizedBox (um “espaçador” de Widgets), para deixar um espaço entre o número aleatório e o botão gerador.

Privacy notice Send feedback 3 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

RUN

```
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
      )
    );
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

E criamos um botão a partir do Widget RaisedButton

DartPad | New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```
      child: Text(
        "Gerador de Números",
        style: TextStyle(fontSize: 28)
      )
    );

  }

}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Realizamos uma instância da classe de estado na classe que representa a GUI de geração de número para interligar o Widget ao seu estado, a partir do método createState().

Privacy notice Send feedback | 1 issues show | Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

Exercício:

Insira o Widget NumeroAleatorio recém-criado na página principal da aplicação;

DartPad | New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

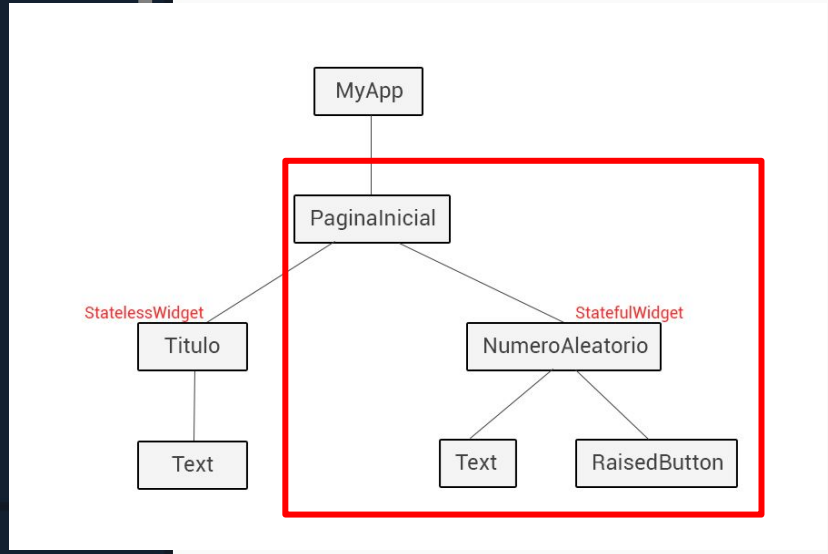
```
}

class PaginaInicial extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(
        title: Text("Gerador aleatório")
      ),
      body: Center(
        child: Column(
          children: [
            Titulo(),
            SizedBox(height: 30),
            NumeroAleatorio()
          ]
        )
      )
    );

  }

}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
```

Gerador aleatório

Gerador de Números

0

Gerar número



Console 1 | Documentation

Privacy notice | Send feedback | 1 issues show | Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

RUN

```
}

class PaginaInicial extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(
        title: Text("Gerador aleatório")
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Titulo(),
            SizedBox(height: 30),
            NumeroAleatorio()
          ]
        )
      )
    );

  }

}

class Titulo extends StatelessWidget {

  @override
  Widget build(BuildContext context) {
```

Console Documentation

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```
  }

}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
      )
    );

  }

}
```

Console Documentation

Gerador aleatório

Gerador de Números

Ainda precisamos dizer que algo será feito quando o usuário tocar no botão “Gerar número”.

Precisamos implementar um evento!

Privacy notice Send feedback

1 issues show

Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

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

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: PaginaInicial()
    );

  }
}

class PaginaInicial extends StatelessWidget {

  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(
        title: Text("Gerador aleatório")
      ),
      body: Center(
        child: Column(
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Precisamos importar a biblioteca math do Dart para que possamos gerar números aleatórios.

Privacy notice Send feedback 2 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad New Pad Reset Format Install SDK calm-wind-6600 Samples

RUN

```
}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
      )
    );

  }

}
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Na classe de estado, declaramos uma variável que representa o número aleatório corrente.

Privacy notice Send feedback 2 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad <> New Pad | Reset | Format | Install SDK | calm-wind-6600 | Samples

RUN

```
}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  void gerarNumero() {

  }

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
      )
    );
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Implementamos um método para representar o evento gerador de números aleatórios.

Privacy notice Send feedback 2 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

DartPad New Pad Reset Format Install SDK calm-wind-6600 Samples

RUN

```
}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  void gerarNumero() {
    setState(() {

    });
  }

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
        ]
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

Adicionamos a callback setState para que o bloco de comandos do método seja apto para alterar estados de elementos do meu app.

Privacy notice Send feedback 2 issues show Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

RUN

```
}

class NumeroAleatorio extends StatefulWidget {

  @override
  NumeroAleatorioState createState() => NumeroAleatorioState();

}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  void gerarNumero() {
    setState(() {
      Random numeroAleatorio = new Random();
      numero = numeroAleatorio.nextInt(1000);
    });
  }

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            child: Text("Gerar número")
          )
```

Console 1 Documentation

Gerador aleatório

Inserimos o bloco de código que permitirá gerar números aleatórios.

O evento está pronto! Mas como dizer que o mesmo deverá ser executado quando o usuário tocar no botão?

DartPad New Pad Reset Format Install SDK calm-wind-6600 Samples

RUN

```
}

class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  void gerarNumero() {
    setState(() {
      Random numeroAleatorio = new Random();
      numero = numeroAleatorio.nextInt(1000);
    });
  }

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("0", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            onPressed: gerarNumero,
            child: Text("Gerar número")
          )
        ]
      )
    );

  }
```

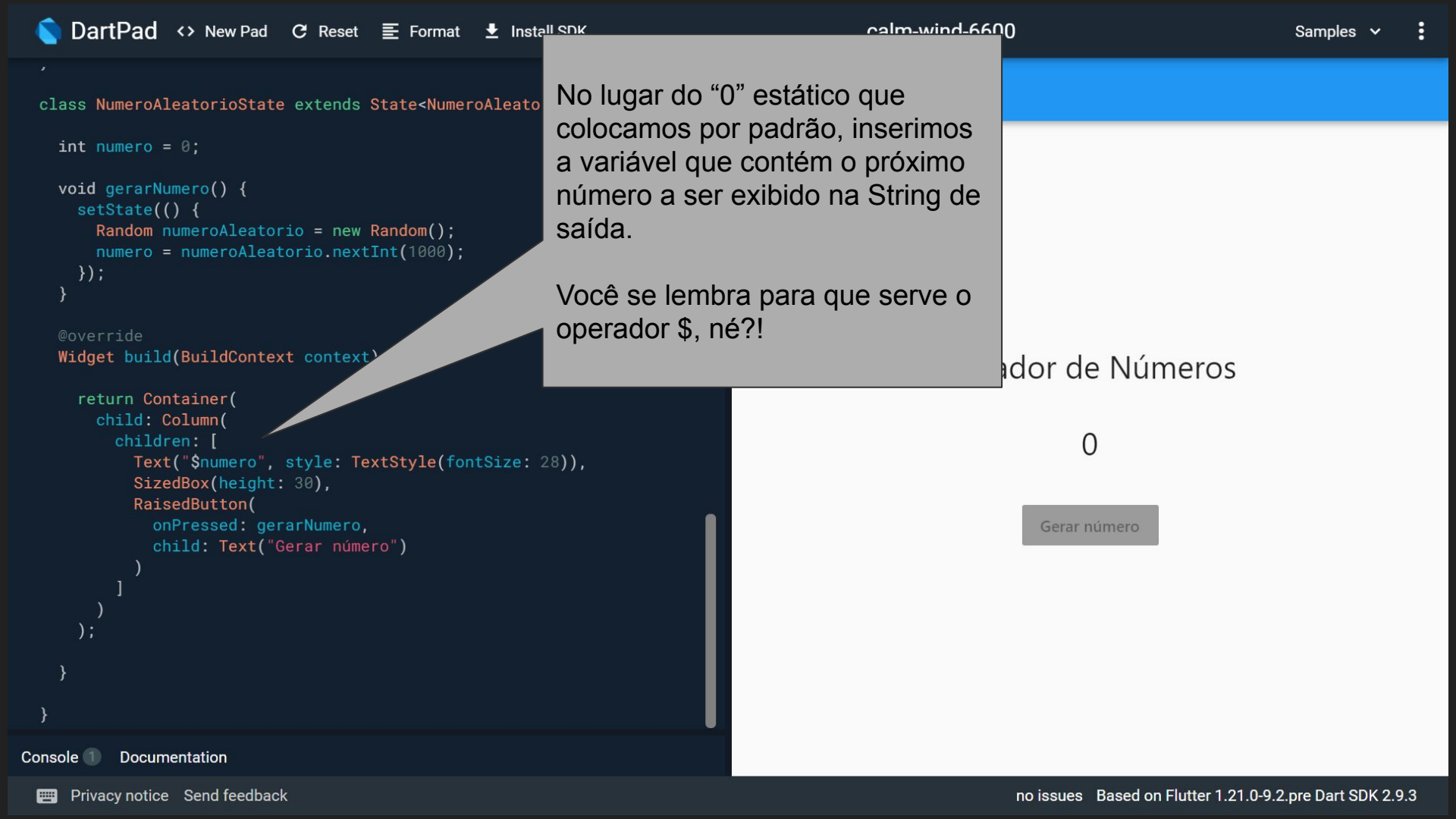
Console 1 Documentation

Gerador aleatório

Adicionamos a propriedade “onPressed” no botão de geração de números, indicando como seu valor o nome do método a ser disparado!

Mas como exibir o conteúdo da variável número no texto criado para tal?

Privacy notice Send feedback no issues Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3


DartPad
New Pad
Reset
Format
Install SDK
calm-wind-6600
Samples
class NumeroAleatorioState extends State<NumeroAleato
int numero = 0;
void gerarNumero() {
setState(() {
Random numeroAleatorio = new Random();
numero = numeroAleatorio.nextInt(1000);
});
}
@override
Widget build(BuildContext context)
return Container(
child: Column(
children: [
Text("$numero", style: TextStyle(fontSize: 28)),
SizedBox(height: 30),
RaisedButton(
onPressed: gerarNumero,
child: Text("Gerar número")
)
]
)
);
}
}
No lugar do “0” estático que colocamos por padrão, inserimos a variável que contém o próximo número a ser exibido na String de saída.
Você se lembra para que serve o operador $, né?!
ador de Números
0
Gerar número
Console 1
Documentation
Privacy notice
Send feedback
no issues
Based on Flutter 1.21.0-9.2.pre Dart SDK 2.9.3

RUN

```
class NumeroAleatorioState extends State<NumeroAleatorio> {

  int numero = 0;

  void gerarNumero() {
    setState(() {
      Random numeroAleatorio = new Random();
      numero = numeroAleatorio.nextInt(1000);
    });
  }

  @override
  Widget build(BuildContext context) {

    return Container(
      child: Column(
        children: [
          Text("$numero", style: TextStyle(fontSize: 28)),
          SizedBox(height: 30),
          RaisedButton(
            onPressed: gerarNumero,
            child: Text("Gerar número")
          )
        ]
      )
    );

  }

}
```

Console 1 Documentation

Gerador aleatório

Gerador de Números

468

Gerar número

Exercício:

Aproveite a base do app de geração de números aleatórios para criar um novo app. Dessa vez, o novo app deverá funcionar como um contador.

Exercício:

Aproveite a base do app anterior para criar um novo app. Dessa vez, o novo app, além de gerar e apresentar um número aleatório, deverá informar se o número gerado é par ou ímpar.