

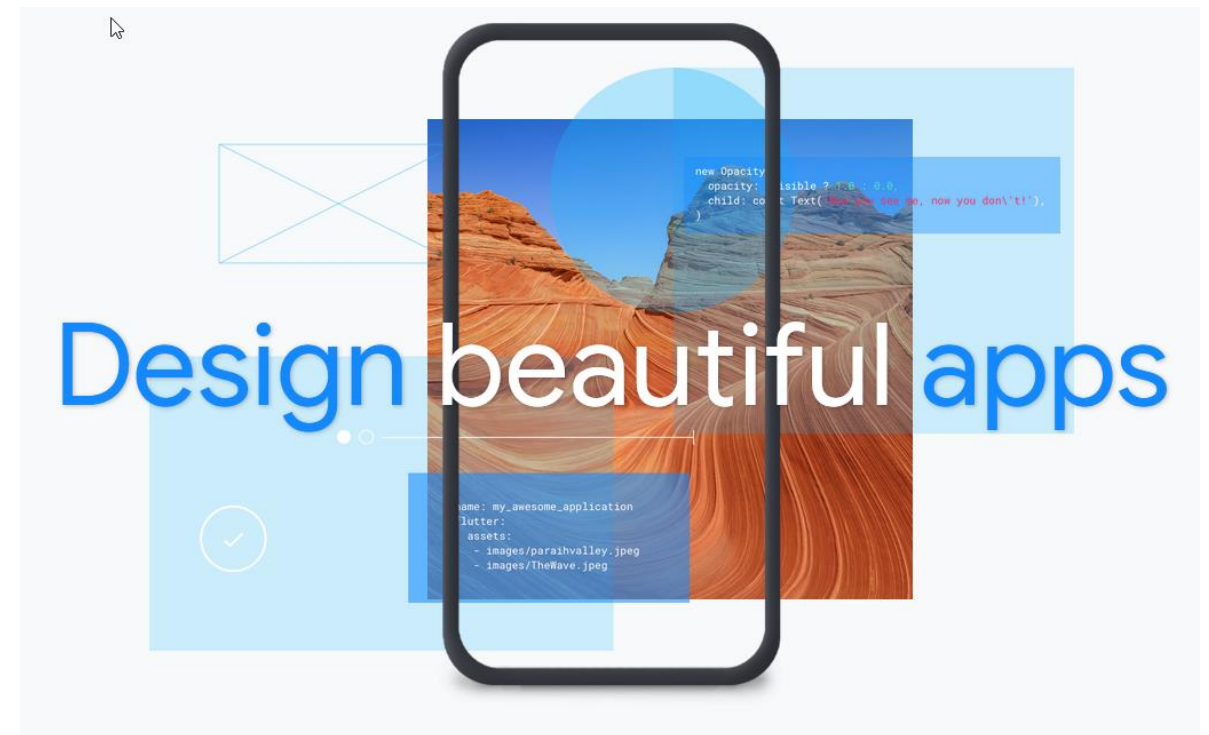


Transformando o futuro das pessoas  
e as pessoas para o futuro.

**#Senacfaz75**



# Desenvolvimento Mobile: Flutter



## O que é o Flutter?

Flutter é um kit de desenvolvimento de interface de usuário (UI toolkit), de código aberto, criado pelo Google, que possibilita a criação de aplicativos compilados nativamente. Atualmente pode compilar para Android, iOS, Windows, Mac, Linux, Google \*Fuchsia e Web.

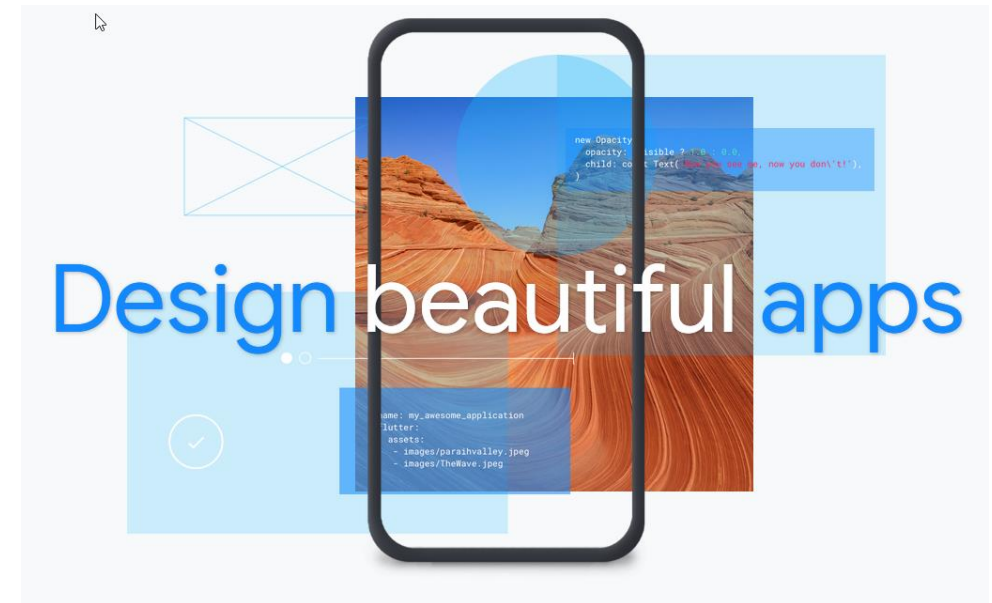
### Características

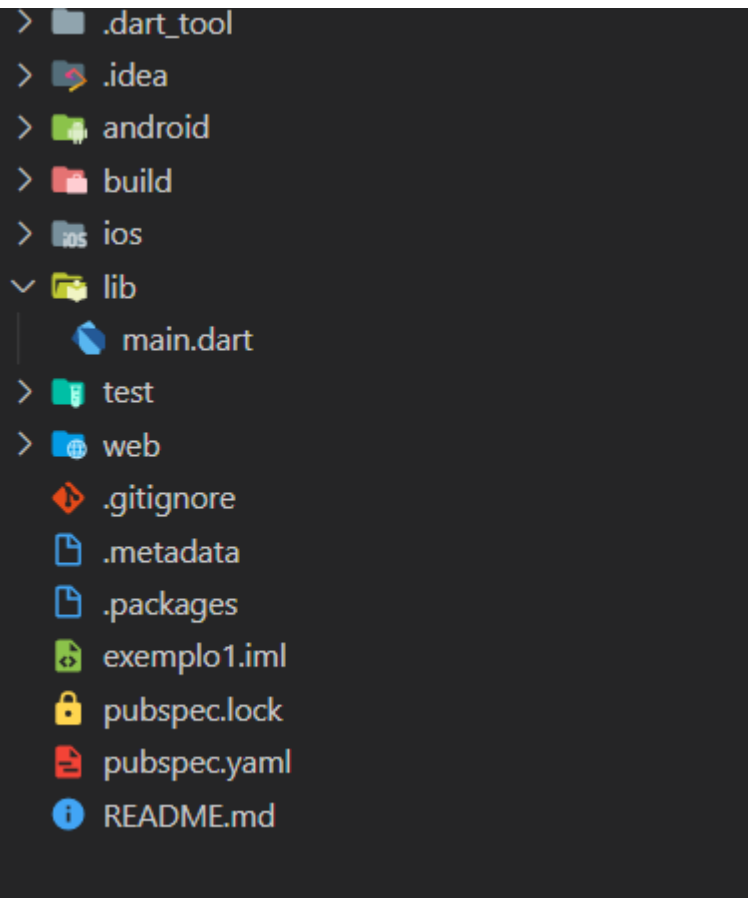
- tem como linguagem base o Dart
- É orientado a objetos
- Misto de Java, Javascript e C#
- Focada em Frontend
- Árvore de Widgets
- A tela inteira é um widget

### Composição

**SDK:** Kit de desenvolvimento para compilar código fonte para código nativo

**Widget:** Componentes UI reutilizáveis desenvolvida do funções, classes e pacotes.





```
> .dart_tool
> .idea
> android
> build
> ios
▼ lib
  main.dart
> test
> web
.gitignore
.metadata
.packages
exemplo1.iml
pubspec.lock
pubspec.yaml
README.md
```

**.dart\_tool:** Criado a partir de um pacote build runner, ficam os arquivos compilados da aplicação. (não mexer) está marcada pelo git como ignorada.

**.idea:** configuração do IntelliJ idea (não mexer).

**Android:** Todos arquivos relacionados a construção do projeto Android.

**build:** componentes de construção do projeto e compilados.

**ios:** pasta de projeto para o XCode.

**lib:** pasta onde programamos nosso projeto, criações de widgets, start do nosso projeto.

**test:** para que for trabalhar com testes, pode-se excluir essa pasta.

**web:** Pasta nova.

**.gitignore:** arquivos que serão ignorados no comit do git.

**.metadata:** gerenciados pelo Flutter.

**.packages:** rotas de dependência.

**.iml:** criado para intellij.

**pubspec.lock e pubspec.yaml:** Criados juntos para configuração do App, exemplo caminhos de assets, dependências.

## Primeiro programa

**Import:** para importar bibliotecas e arquivos externos

**Main():** Classe principal da aplicação Flutter

**runApp():** Disparador do aplicativo

**MaterialApp():** Classe de desenvolvimento visual do Flutter

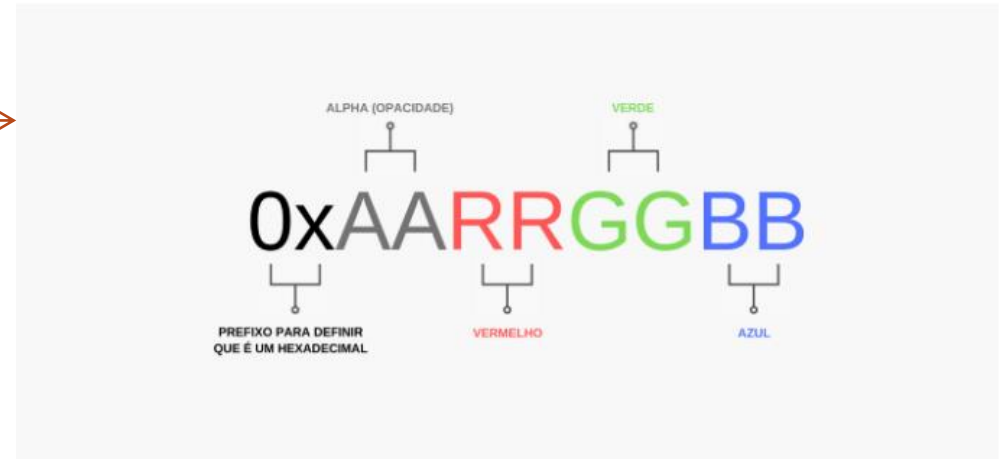
**Title e home:** Parâmetros nomeados de MaterialApp()

**Container():** “Envelope” para um conteúdo da aplicação, lembra uma div.

**Color(minúsculo):** parâmetro nomeado para inserção de Cores

**Colors(maiúsculo):** Construtor para aplicação da cor

```
main01_primeiroPrograma.dart X
lib > main01_primeiroPrograma.dart > ...
1  import 'package:flutter/material.dart';
2
   Run | Debug | Profile
3  main() => runApp(MaterialApp(
4    title: 'Sorteio da MegaSena',
5    home: Container(
6      color: Colors.white,
7    ), // Container
8  )); // MaterialApp
```



<https://api.flutter.dev/flutter/dart-ui/Color-class.html>

## Inserindo Colunas

**Import:** para importar bibliotecas e arquivos externos

**Main():** Classe principal da aplicação Flutter

**runApp():** Disparador do aplicativo

**MaterialApp():** Classe de desenvolvimento visual do Flutter

**Title e home:** Parâmetros nomeados de MaterialApp()

**Container():** “Envelope” para um conteúdo da aplicação, lembra uma div.

**Color(minúsculo):** parâmetro nomeado para inserção de Cores

**Colors(maiúsculo):** Construtor para aplicação da cor

**Child:** Parâmetro filho de Container(), só recebe 1 widget

**Column():** Widget para inserir colunas

**Children:** Array filho para vários Widget

**Text():** Widget para inserir textos.

```
lib > main02_colunas.dart > ...
1 //Inserindo um Widget Coluna
2 import 'package:flutter/material.dart';
3
4 Run | Debug | Profile
5 main() => runApp(MaterialApp(
6   title: 'Sorteio da MegaSena',
7   home: Container(
8     color: Colors.grey,
9     child: Column(
10      //Uso um children para colocar vários widgets
11      children: [
12        Text('Texto 1'),
13        Text('Texto 2'),
14        Text('Texto 3')],
15    ), // Column
16  )); // Container // MaterialApp
```

## Inserindo Linhas

**Import:** para importar bibliotecas e arquivos externos

**Main():** Classe principal da aplicação Flutter

**runApp():** Disparador do aplicativo

**MaterialApp():** Classe de desenvolvimento visual do Flutter

**Title e home:** Parâmetros nomeados de MaterialApp()

**Container():** “Envelope” para um conteúdo da aplicação, lembra uma div.

**Color(minúsculo):** parâmetro nomeado para inserção de Cores

**Colors(maiúsculo):** Construtor para aplicação da cor

**Child:** Parâmetro filho de Container(), só recebe 1 widget

**Row():** Widget para inserir linhas

**Children:** Array filho para vários Widget

**Text():** Widget para inserir textos.

```
lib > main03_linhas.dart > ...
1 //Inserindo um Widget Linhas
2 import 'package:flutter/material.dart';
3
4 Run | Debug | Profile
5 main() => runApp(MaterialApp(
6   title: 'Sorteio da MegaSena',
7   home: Container(
8     color: Colors.grey,
9     child: Row(
10      //Uso um children para colocar vários widgets
11      children: [
12        Text('Texto 1'),
13        Text('Texto 2'),
14        Text('Texto 3')],
15    ), // Row
16  )); // Container // MaterialApp
```

## Formatando textos

- **style:** Definindo as formatações do Texto
- **fontSize:** Tamanho da fonte
- **fontStyle:** Estilo da fonte
- **fontWeight:** Peso da fonte (Negrito)
- **letterSpacing:** Espaçamento entre os caracteres
- **decoration:** sublinhados e tachados no texto
- **color:** Cor da fonte

```
lib > main04_FormatacaoTexto.dart > ...
1 //Formatação de texto
2 import 'package:flutter/material.dart';
3
4 Run | Debug | Profile
5 main() => runApp(MaterialApp(
6   title: 'Sorteio MegaSena',
7   //container é um envólucro para os conteúdos
8   home: Container(
9     color: Colors.white,
10
11     //Definindo um filho pra o contaier
12     child: Column(
13       //Definindo um children para o Column
14       children: [
15         Text(
16           'Esse é um exemplo de texto extenso para o widget',
17           style: TextStyle(
18             //Definindo as formatações do Texto
19             fontSize: 40,
20             fontStyle: FontStyle.normal,
21             fontWeight: FontWeight.normal,
22
23             //Espaçamento entre os caracteres
24             letterSpacing: 2,
25
26             //Retirar o sublinhado
27             decoration: TextDecoration.none,
28
29             //Aplicar na Fonte
30             color: Colors.black
31           ), // TextStyle
32         ), // Text
33       ],
34     ), // Column
35   ), // Container
36 )); // MaterialApp
```



## Espaçamentos internos e externos (Padding e Margin)

### padding: EdgeInsets

- .all : Espaçamento para todos os lados (valor único)
- .fromLTRB: Espaçamento para todos os lados (left, top, right, bottom)
- .only: Espaçamento individual (left: 10, por exemplo)

### margin: EdgeInsets

- .all : Margem para todos os lados (valor único)
- .fromLTRB: Margem para todos os lados (left, top, right, bottom)
- .only: Espaçamento individual (left: 10, por exemplo)

**BoxDecoration:** Construtor para as bordas do box

**Color:** Aplicação de cores

**Border:** Definindo a borda

**Width:** Espessura

**Color:** Cor



```

lib > main.dart > main
1 import 'package:flutter/material.dart';
  Run | Debug | Profile
2 main() => runApp(MaterialApp(
3   //Removendo a faixa de debug
4   debugShowCheckedModeBanner: false,
5   title: 'Sorteio MegaSena',
6   home: Container(
7     //Defindo espaçamento internos
8     //.all .fromLTRB .only
9     //padding: EdgeInsets.all()
10
11     padding: EdgeInsets.fromLTRB(20, 50, 20, 0),
12     //Definindo os espaçamento externos
13     //.all .fromLTRB .only
14     //margin: EdgeInsets.all()
15
16     margin: EdgeInsets.fromLTRB(20, 50, 20, 20),
17     //Borda e cor do Container
18
19     decoration: BoxDecoration(
20       //Adicionando uma cor de background
21       color: Colors.blueGrey,
22
23       //Adicionando uma borda
24       border: Border.all(
25         width: 5,
26         color: Colors.amber
27       ), // Border.all
28     ), // BoxDecoration
29

```

```

30   //Adicionar um texto dentro do Container
31   child: Column(
32     children: [
33       Text(
34         'Texto exemplo para visualizar o preenchimento da tela do App',
35         textAlign: TextAlign.center,
36         style: TextStyle(
37           fontSize: 25,
38           fontWeight: FontWeight.normal,
39           decoration: TextDecoration.none,
40           color: Colors.white
41         ), // TextStyle
42       ) // Text
43     ],
44   ), // Column
45 ), // Container
46 ) // MaterialApp
47 );

```

# Espaçamento no Widget

```
lib > main.dart > main
1 //Padding no Widget
2 import 'package:flutter/material.dart';
3
4 Run | Debug | Profile
5 main() => runApp(MaterialApp(
6   //Removendo a faixa de debug
7   debugShowCheckedModeBanner: false,
8   title: 'Padding no Widget',
9   home: Container(
10    decoration: BoxDecoration(
11     color: Colors.blueGrey,
12     border: Border.all(width: 5, color: Colors.white)), // BoxDecoration
13    child: Row(
14     children: [
15      Text(
16       'Texto1',
17       style: TextStyle(
18        fontSize: 20,
19        decoration: TextDecoration.none,
20        color: Colors.redAccent), // TextStyle
21      ), // Text
22
23      //Padding diretamente no widget
24      Padding(
25       padding: EdgeInsets.all(25),
26       child: Text(
27        'Texto 2',
28        style: TextStyle(
29         fontSize: 20,
30         decoration: TextDecoration.none,
31         color: Colors.blueAccent
32       ), // TextStyle
33     ), // Text
34   ), // Padding
35
```

```
36
37   Text(
38     'Texto 3',
39     style: TextStyle(
40      fontSize: 20,
41      decoration: TextDecoration.none,
42      color: Colors.amberAccent
43    ), // TextStyle
44  ), // Text
45 ], // Row
46 ) // Container
47 )); // MaterialApp
```

→ O construtor `Padding()` pode ser usado diretamente em outros widgets.

Há uma forma rápida para colocar um padding no widget, basta teclemos `<CTRL> + <.>`. Assim podemos envolver nosso widget em um `Padding`



# Alinhamentos

## Alinhamento Principal (`mainAxisAlignment`: `MainAxisAlignment`)

**`mainAxisAlignment`:** `MainAxisAlignment.start`: Alinhamento no início

**`mainAxisAlignment`:** `MainAxisAlignment.end`: Alinhamento no Fim

**`mainAxisAlignment`:** `MainAxisAlignment.center`: Alinhamento no centro

**`mainAxisAlignment`:** `MainAxisAlignment.spaceEvenly`: Alinhamento distribuído

## Rebatendo o alinhamento do Eixo Principal (`crossAxisAlignment`)

**`crossAxisAlignment`:** `CrossAxisAlignment.start`: Alinhamento no início

**`crossAxisAlignment`:** `CrossAxisAlignment.end`: Alinhamento no Fim

**`crossAxisAlignment`:** `CrossAxisAlignment.center`: Alinhamento no centro



# Alinhamentos

```
lib > main.dart > main
1 //Alinhamentos
2
3 import 'package:flutter/material.dart';
4
5 Run | Debug | Profile
6 main() => runApp(MaterialApp(
7   //removendo a faixa de debug
8   debugShowCheckedModeBanner: false,
9
10  title: 'Alinhamentos',
11  home: Container(
12    //definindo margens
13    margin: EdgeInsets.only(top: 60),
14
15    //definindo borda e cor do container
16    decoration: BoxDecoration(
17      border: Border.all(
18        width: 5,
19        color: Colors.blue
20      ) // Border.all
21    ), // BoxDecoration
22
23    //Criando um child
24    child: Row(
25      children: [
26
27        //Criando textos
28        Text(
29          'T1',
30          style: TextStyle(
31            decoration: TextDecoration.none,
32            fontSize: 30
33          ), // TextStyle
34        ), // Text
```

```
35      //Espaçando internamente o T2
36      Text(
37        'T2',
38        style: TextStyle(
39          decoration: TextDecoration.none,
40          fontSize: 30
41        ), // TextStyle
42      ), // Text
43
44      Text(
45        'T3',
46        style: TextStyle(
47          decoration: TextDecoration.none,
48          fontSize: 30
49        ), // TextStyle
50      ) // Text
51    ],
52
53    //Alinhamento principal
54    //mainAxisAlignment: MainAxisAlignment.start .center .end .spaceEvenly
55    mainAxisAlignment: MainAxisAlignment.start,
56
57    //Configurar o eixo que cruza o alinhamento principal
58    //crossAxisAlignment: CrossAxisAlignment.start .end .center
59    crossAxisAlignment: CrossAxisAlignment.start,
60  ), // Row
61 ), // Container
62 )); // MaterialApp
```



Siga o Senac em Minas nas Redes Sociais:

