Text

O Widget Text, como já vimos, trata-se do Widget responsável por introduzir um texto em nossa aplicação.

Esse Widget possui algumas propriedades importantes que devem ser conhecidas por qualquer desenvolvedor Flutter. Vejamos quais são elas.

style

Essa talvez seja a propriedade mais utilizada do Widget Text. Como o próprio nome já sugere, ela nos permite estilizar o texto.

Essa propriedade deve receber um objeto TextStyle. Dentre as possíveis customizações encontramos:

- **color;**
- **decoration;**
- **fontFamily;**
- **fontSize;**
- **fontStyle;**
- **fontWeight.**

color

Talvez essa seja a propriedade mais utilizada na estilização de textos, e como o próprio nome já sugere, nos permite alterar a cor do texto.

Para representarmos uma cor, podemos utilizar duas estratégias:

- Utilizamos uma das constantes da classe Colors, nativa do Flutter;
- Declaramos nossa cor personalizada, utilizando o construtor da classe Color e passando a cor em Hexadecimal.

```dart
Text("Esse texto é verde!",
  style: TextStyle(
    color: Colors.green,
  ),
),
```

decoration

Essa propriedade recebe um objeto TextDecoration, nos permitindo decorar nosso texto, colocando um sublinhado, tachado ou uma linha superior no mesmo.

```dart
Text("Esse texto é verde!",
  style: TextStyle(
    decoration: TextDecoration.underline,
  ),
),
```

fontFamily

Essa propriedade recebe uma string contendo o nome da família de fonte utilizada no texto.

É importante ressaltar que a família de fontes deve estar devidamente declarada no arquivo pubspec.yaml para que essa possa ser usada.

```dart
Text("Usando a família de fontes Open Sans!",
  style: TextStyle(
    fontFamily: "Open Sans",
  ),
),
```

fontSize

Essa propriedade recebe um valor double, o qual irá representar o tamanho da fonte do texto em questão.

```
Text("Um texto qualquer",
  style: TextStyle(
    fontSize: 20.0,
  ),
),
```

fontStyle

Essa propriedade recebe um objeto FontStyle. Para tal utilizamos uma constante da própria classe FontStyle. Dentre as opções temos:

- **italic:** Fonte com estilo itálico;
- **normal:** Fonte com estilo regular.

```
Text("Um texto em italico",
  style: TextStyle(
    fontStyle: FontStyle.italic,
  ),
),
```

fontWeight

Essa propriedade recebe um objeto fontWeight. Para tal utilizamos uma constante da própria classe fontWeight. Dentre as opções temos:

- **bold:** Fonte com estilo negrito;
- **normal:** Fonte com estilo regular.

```
Text("Um texto em negrito",
  style: TextStyle(
    fontWeight: FontWeight.bold,
  ),
),
```

textAlign

Como o próprio nome já sugere, essa propriedade nos permite alinhar o texto.

Essa propriedade deve receber um objeto TextAlign. Dentre as possíveis customizações encontramos:

- **center:** Alinhado no centro;
- **end:** Alinhado no final;
- **justify:** Alinhamento justificado;
- **left:** Alinhado a esquerda;
- **right:** Alinhado à direita;
- **start:** Alinhado no início.