

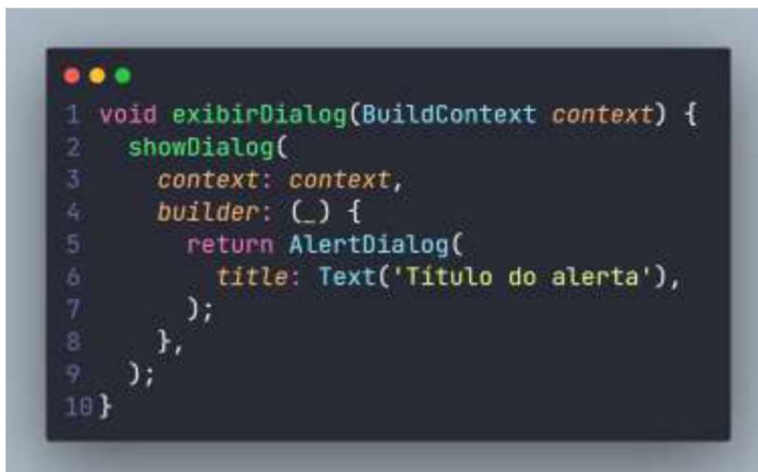
AlertDialog

Como o próprio nome já sugere, o *Widget AlertDialog* exibe uma janela de Diálogo alertando o usuário de alguma decisão importante, a qual necessitamos de sua confirmação.

Esse *Widget* é muito útil em casos que necessitamos solicitar que o usuário confirme estar ciente de alguma ação potencialmente perigosa ou importante, para a qual necessitamos de sua confirmação.

title

A propriedade *title*, como o próprio nome já sugere, é a propriedade que recebe o texto da caixa de alerta. Essa propriedade recebe um *Widget*, o qual normalmente é um *Widget Text*.

A screenshot of a code editor with a dark background and light-colored text. The code is in Dart and defines a function named `exibirDialog` that takes a `BuildContext` parameter. Inside the function, `showDialog` is called with `context: context` and a `builder` function. The `builder` function returns an `AlertDialog` widget with a `title` property set to `Text('Título do alerta')`. The code is as follows:

```
1 void exibirDialog(BuildContext context) {  
2   showDialog(  
3     context: context,  
4     builder: (_) {  
5       return AlertDialog(  
6         title: Text('Título do alerta'),  
7       );  
8     },  
9   );  
10 }
```

titlePadding

A propriedade *titlePadding*, como o próprio nome já sugere, define o espaço entre o texto do título e a borda da caixa de alerta.

Essa propriedade deve receber um objeto *EdgeInsetsGeometry*. Normalmente utilizamos uma constante provida da classe *EdgeInsets*, que herda *EdgeInsetsGeometry*.

```

1 void exibirDialog(BuildContext context) {
2   showDialog(
3     context: context,
4     builder: (_) {
5       return AlertDialog(
6         title: Text('Título do alerta'),
7         titlePadding: EdgeInsets.all(40),
8       );
9     },
10  );
11}

```

titleTextStyle

A propriedade *titleTextStyle*, como o próprio nome já sugere, define o estilo do título da caixa de alerta.

Essa propriedade deve receber um objeto *TextStyle*, onde podemos definir propriedades como o tamanho e a cor do texto.

content

A propriedade *content*, como o próprio nome já sugere, define o conteúdo da caixa de alerta.

Essa propriedade deve receber um *Widget*, sendo assim podemos criar um conteúdo totalmente personalizado para nossa caixa de alerta.

```

1 void exibirDialog(BuildContext context) {
2   showDialog(
3     context: context,
4     builder: (_) {
5       return AlertDialog(
6         title: Center(
7           child: Text('Remoção da galeria'),
8         ),
9         content: Text("Remover a foto da galeria?"),
10      );
11    },
12  );
13}

```

contentPadding

A propriedade *contentPadding*, define o espaço entre o conteúdo e a borda da caixa de alerta.

Essa propriedade deve receber um objeto *EdgeInsetsGeometry*. Normalmente utilizamos uma constante provida da classe *EdgeInsets*, que herda *EdgeInsetsGeometry*.

A screenshot of a code editor with a dark background and light-colored text. The code is in Dart and defines a function named `exibirDialog` that takes a `BuildContext` parameter. Inside the function, `showDialog` is called with a builder that returns an `AlertDialog`. The `AlertDialog` has a title `Center`, a child `Text` widget with the text 'Remoção da galeria', and a content `Text` widget with the text 'Remover a foto da galeria?'. The `contentPadding` property of the `AlertDialog` is set to `EdgeInsets.all(40.0)`.

```
1 void exibirDialog(BuildContext context) {  
2   showDialog(  
3     context: context,  
4     builder: (_) {  
5       return AlertDialog(  
6         title: Center(  
7           child: Text('Remoção da galeria'),  
8         ),  
9         content: Text("Remover a foto da galeri  
10        a?"),  
11         contentPadding: EdgeInsets.all(40.0),  
12       );  
13     },  
14   );  
}
```

contentTextStyle

A propriedade *contentTextStyle*, define o estilo do conteúdo da caixa de alerta.

Essa propriedade deve receber um objeto *TextStyle*. Onde podemos definir propriedades como o tamanho e a cor do texto.

```

1 void exibirDialog(BuildContext context) {
2   showDialog(
3     context: context,
4     builder: (_) {
5       return AlertDialog(
6         title: Center(
7           child: Text('Remoção da galeria'),
8         ),
9         content: Text("Remover a foto da galeri
10      a?"),
11         contentPadding: EdgeInsets.all(40.0),
12         contentTextStyle: TextStyle(
13           color: Colors.red,
14         ),
15       );
16     },
17   );
18 }

```

actions

A propriedade *actions*, como o próprio nome já sugere, define as ações possíveis que o usuário pode escolher na caixa de alerta. Normalmente são duas opções, uma para confirmação e outra para o cancelamento.

Essa propriedade recebe uma lista de *Widgets*. Normalmente utilizamos botões (*ElevatedButton* ou *TextButton*).

```

1 void exibirDialog(BuildContext context) {
2   showDialog(
3     context: context,
4     builder: (_) {
5       return AlertDialog(
6         title: Center(
7           child: Text('Remoção da galeria'),
8         ),
9         content: Text("Remover a foto da galeri
10        a?"),
11         actions: [
12           TextButton(
13             child: Text(
14               "Sim",
15               style: TextStyle(
16                 color: Colors.green,
17               ),
18             ),
19             onPressed: () {
20               print("Clicou em sim!!!");
21               Navigator.pop(context);
22             },
23           ),
24           ElevatedButton(
25             style: ElevatedButton.styleFrom(
26               primary: Colors.red,
27             ),
28             child: Text(
29               "Não",
30             ),
31             onPressed: () {
32               print("Clicou em não!!!");
33               Navigator.pop(context);
34             },
35           ),
36         ],
37       );
38     },
39   );
40 }

```

actionsPadding

A propriedade *actionsPadding*, como o próprio nome já sugere, define o espaço entre os *Widgets* da lista e a borda da caixa de alerta.

Essa propriedade deve receber um objeto *EdgeInsetsGeometry*. Normalmente utilizamos uma constante provida da classe *EdgeInsets*, que herda *EdgeInsetsGeometry*.

```
1 void exibirDialog(BuildContext context) {
2   showDialog(
3     context: context,
4     builder: (_) {
5       return AlertDialog(
6         title: Center(
7           child: Text('Remoção da galeria'),
8         ),
9         content: Text("Remover a foto da galeri
10        a?"),
11         actions: [
12           // Botões
13         ],
14         actionsPadding: EdgeInsets.all(10.0),
15       );
16     },
17   );
18 }
```