

Card

O *Widget Card* permite a criação de um container visual com as bordas arredondadas. Por padrão esse *Widget* possui a cor de fundo branca. Mas o mesmo pode ser alterado.

Propriedade Child

A propriedade *child* é uma das principais propriedades deste *Widget*. Como o próprio nome já sugere, define quem será o *Widget* filho deste.

```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13    ),  
14  ),  
15)
```

Propriedade color

A propriedade *color*, como o próprio nome já sugere, nos permite controlar a cor do ícone que será apresentado. Essa propriedade receberá um objeto *Color*. Normalmente utilizamos umas das constantes previstas na classe *Colors*. Mas podemos utilizar uma cor personalizada, se desejarmos.

```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13      color: Colors.green[300],  
14    ),  
15  ),  
16)
```

Propriedade elevation

A propriedade *elevation* é utilizada para dar ênfase ao *Widget*. Essa propriedade cria uma sombra ao redor do *Widget*.

```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13      color: Colors.green[300],  
14      elevation: 10.0,  
15    ),  
16  ),  
17)
```

Propriedade shadowColor

A propriedade *shadowColor* é utilizada para dar cor à sombra criada ao redor do *Widget*.

```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13      color: Colors.green[300],  
14      elevation: 10.0,  
15      shadowColor: Colors.greenAccent,  
16    ),  
17  ),  
18)
```

Propriedade *margin*

A propriedade *margin*, como o próprio nome já sugere, define uma margem entre o card e seu Widget pai, dessa forma o *card* e seu filho serão empurrados mais para dentro da tela.

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

- ***all***: Um mesmo valor para todas as 4 direções;
- ***fromLTRB***: Um valor diferente para cada uma das 4 direções;
- ***only***: Apenas nas direções definidas como parâmetro;
- ***symmetric***: Um valor para topo e inferior, outro para esquerda e direita.



```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13      color: Colors.green[300],  
14      elevation: 10.0,  
15      shadowColor: Colors.greenAccent,  
16      margin: EdgeInsets.all(50.0),  
17    ),  
18  ),  
19)
```

Propriedade *shape*

A propriedade *shape* nos permite definir o arredondamento das bordas do *card*. Essa propriedade recebe um objeto *ShapeBorder* ou um descendente dessa classe.

Normalmente utilizamos um objeto *RoundedRectangleBorder*, mas também é possível utilizar um objeto *ContinuousRectangleBorder* ou *CircleBorder*.

```
1 Scaffold(  
2   appBar: AppBar(  
3     title: Text("Aplicação Exemplo"),  
4   ),  
5   body: Container(  
6     child: Card(  
7       child: Container(  
8         alignment: Alignment.center,  
9         width: double.infinity,  
10        height: 100.0,  
11        child: Text("Conteúdo"),  
12      ),  
13      color: Colors.green[300],  
14      elevation: 10.0,  
15      shadowColor: Colors.greenAccent,  
16      margin: EdgeInsets.all(50.0),  
17      shape: RoundedRectangleBorder(  
18        borderRadius: BorderRadius.circular(30.0),  
19      ),  
20    ),  
21  ),  
22)
```