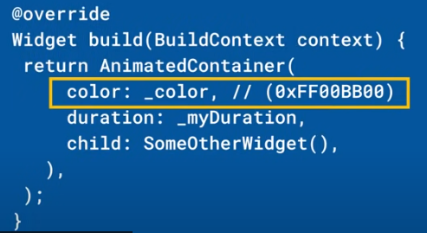
**AnimatedContainer**

Es un Widget para hacer una animación implícita.

Ejemplo: lo creamos con un color



**Y al regenerarlo con color diferente:**



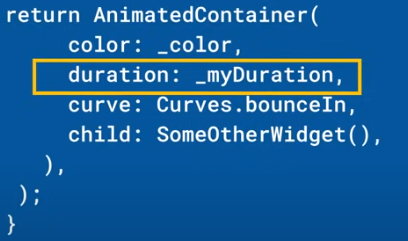
lo hace con una animación que interpola linealmente entre ambos colores,

Podemos animar

* Contornos
* Radios de las esquinas
* Imágenes de fondo
* Sombras
* Gradientes
* Formas
* Rellenado
* Ancho
* Altura
* Alineación
* Transformaciones

La duración se controla con el parámetro duration.

También se le puede asignar un Curve



|  |  |
| --- | --- |
| **Código** | **Comentarios** |
| import 'dart:math' show Random;  import 'package:flutter/material.dart';  class AnimatedScreen extends StatefulWidget {    const AnimatedScreen({Key? key}) : super(key: key);    @override    State<AnimatedScreen> createState() => \_AnimatedScreenState();  }  class \_AnimatedScreenState extends State<AnimatedScreen> {    double \_width = 50;    double \_height = 50;    Color \_color = Colors.indigo;    Color \_color2 = Colors.red;    double \_border = 5;    BorderRadiusGeometry \_borderRadius = BorderRadius.circular(10);    @override    Widget build(BuildContext context) {      return Scaffold(        appBar: AppBar(title: const Text('Animated Container')),        body: Center(          child: AnimatedContainer(            height: \_height,            width: \_width,            duration: const Duration(milliseconds: 300),            curve: Curves.easeOutCubic,            decoration: BoxDecoration(                color: \_color,                borderRadius: \_borderRadius,                border: Border.all(width: \_border, color: \_color2)),          ),        ),        floatingActionButton: FloatingActionButton(          child: const Icon(            Icons.play\_circle\_outline,            size: 35,          ),          onPressed: changeShape,        ),      );    }    void changeShape() {      final random = Random();      \_width = random.nextInt(300).toDouble() + 70;      \_height = random.nextInt(300).toDouble() + 70;      \_color = Color.fromRGBO(          random.nextInt(255), random.nextInt(255), random.nextInt(255), 1);      \_color2 = Color.fromRGBO(          random.nextInt(255), random.nextInt(255), random.nextInt(255), 1);      \_borderRadius = BorderRadius.circular(random.nextInt(100).toDouble() + 10);      \_border = random.nextInt(20).toDouble();      setState(() {});    }  } |  |

Apretando el botón animamos el alto, ancho, color, grosor y color del borde, radio de las esquinas:

Imagen que contiene Icono

Descripción generada automáticamenteForma

Descripción generada automáticamente con confianza mediaIcono

Descripción generada automáticamenteIcono

Descripción generada automáticamente

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -