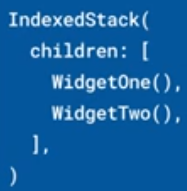
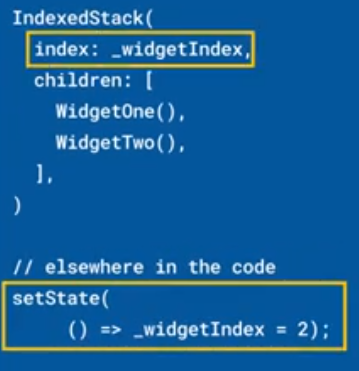
**IndexedStack**

Para pasar de un widget a otro (como si se cambiara de canal en el tele).

Muestra solo un hijo a la vez conservando el estado de todos los hijos.

|  |  |
| --- | --- |
| **Código** | **Comentarios** |
| import 'package:flutter/material.dart';  class IndexedStackScreen extends StatefulWidget {    const IndexedStackScreen({Key? key}) : super(key: key);    @override    State<IndexedStackScreen> createState() => \_IndexedStackScreenState();  }  class \_IndexedStackScreenState extends State<IndexedStackScreen> {    int contador = 0;    int index = 0;    @override    Widget build(BuildContext context) {      return Scaffold(        appBar: AppBar(          title: const Text('IndexedStack'),          centerTitle: true,        ),        body: Center(          child: IndexedStack(            index: index,            children: [              MiWidget(                color: Colors.blue,                text: "$contador",                isCircle: false,              ),              MiWidget(                color: Colors.red,                text: "$contador",                isCircle: true,              ),              MiWidget(                color: Colors.yellow,                text: "$contador",                isCircle: false,              ),              MiWidget(                color: Colors.green,                text: "$contador",                isCircle: true,              )            ],          ),        ),        floatingActionButton: FloatingActionButton(          onPressed: () {            setState(() {              contador++;              index == 3 ? index = 0 : index++;            });          },          child: const Icon(Icons.play\_arrow),        ),      );    }  }  //---------------------------------------------------------------------------------  class MiWidget extends StatelessWidget {    final Color color;    final String text;    final bool isCircle;    const MiWidget(        {super.key,        required this.color,        required this.text,        required this.isCircle});    @override    Widget build(BuildContext context) {      return Container(        width: 170,        height: 170,        decoration: BoxDecoration(            color: color,            borderRadius: isCircle ? BorderRadius.circular(100) : null),        child: Center(          child: Text(text,              style: const TextStyle(                  color: Colors.black,                  fontSize: 28,                  fontWeight: FontWeight.bold)),        ),      );    }  } |  |

Imagen que contiene Gráfico

Descripción generada automáticamente Imagen que contiene Esquemático

Descripción generada automáticamente Gráfico

Descripción generada automáticamente Imagen que contiene Icono

Descripción generada automáticamente Imagen que contiene Gráfico

Descripción generada automáticamente

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