

# Flutter

---

¿Qué es Flutter? Flutter es un SDK portable, open source framework y UI Widget Library, está fuertemente inspirado en react, su objetivo es crear hermosas aplicaciones compiladas de forma nativa para móviles con una sola base de código.

Los Widget son las piezas de una interfaz como los botones, campos de texto, etc. Flutter provee un amplio rango de widgets que pueden ser usados para construir una interfaz de usuario. Otra ventaja es que podemos usar widgets de terceros o incluso crear nuestros propios widgets, empaquetarlos y compartirlos con otros desarrolladores en [pub.dev](https://pub.dev).

Con Flutter tambien podemos crear aplicaciones incluso para la web, escritorio y embebidos.

## Estrucutra de un proyecto Flutter

- **.dart\_tool**: Contiene archivos de configuración para el proyecto.
- **.idea**: Contiene archivos de configuración para el IDE, en el caso de que estemos usando IntelliJ.
- **android**: Contiene el proyecto de Android.
- **ios**: Contiene el proyecto de iOS.
- **lib**: Contiene el código fuente de la aplicación, es en donde escribiremos la mayor parte del código de Flutter.
- **test**: Contiene los archivos de test.
- **web**: Contiene el proyecto de la web.
- **pubspec.yaml**: Contiene la información del proyecto, como el nombre, versión, dependencias, etc.

## Hola mundo en Flutter

Toda aplicación de Flutter tiene en su main una ejecución de un Widget principal, esto comando es `runApp()` y recibe como parámetro el Widget principal de la aplicación.

```
import 'package:flutter/material.dart';

void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Hola mundo',
      home: Scaffold(
        appBar: AppBar(
          title: Text('Hola mundo'),
        ),
        body: Center(
          child: Text('Hola mundo'),
        ),
      ),
    );
  }
}
```

```

    ),
  ),
);
}
}

```

Toda app creada con Flutter tiene como pieza básica los Widgets, estos son funciones que dibujan ciertos componentes específicos en pantalla y que pueden ser modificados o configurados mediante argumentos que les pasemos, por ejemplo:

```
Text('Hola mundo')
```

En este caso el Widget `Text` recibe como argumento el texto que queremos mostrar en pantalla, pero también podemos pasarle otros argumentos como el color, el tamaño, etc.

Un Widget básico y esencial es el `Scaffold`, este es el que nos permite crear una pantalla con un AppBar y un cuerpo, el cual puede ser cualquier otro Widget.

```

import 'package:flutter/material.dart';

class CounterScreen extends StatelessWidget {
  const CounterScreen({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Counter'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: const [
            Text(
              '10',
              style: TextStyle(fontSize: 160, fontWeight: FontWeight.w100),
            ),
            Text(
              'Clicks',
              style: TextStyle(fontSize: 25),
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {},
        child: const Icon(Icons.plus_one),
      ),
    ),
  ),
);

```

```
    );  
  }  
}
```

## El estado de un Widget

El estado de un Widget es la información que este contiene, por ejemplo, el texto que contiene un `Text` o el color de un `Container`. El estado de un Widget puede ser mutable o inmutable, podemos tener estados globales o locales, etc.

Los Widget `Stateful` permiten tener un estado interno y ciclo de vida como su inicialización y destrucción.

```
@override  
State<CounterScreen> createState() => _CounterScreenState();
```

De esta forma declaramos un estado, por supuesto tenemos que cambiar el Widget a uno `Stateless`.

Declaramos un contador de clicks:

```
int clickCounter = 0;
```

Y cada vez que presionemos el botón actualizamos el estado para ver los cambios en este reflejados en la pantalla.

```
floatingActionButton: FloatingActionButton(  
  onPressed: () {  
    setState(() {  
      clickCounter++;  
    });  
  },  
  child: const Icon(Icons.plus_one),  
),
```

Así se vería la pantalla completa nueva:

```
import 'package:flutter/material.dart';  
  
class CounterScreen extends StatefulWidget {  
  const CounterScreen({super.key});  
  
  @override  
  State<CounterScreen> createState() => _CounterScreenState();  
}
```

```

}

class _CounterScreenState extends State<CounterScreen> {
  int clickCounter = 0;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Counter'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Text(
              '$clickCounter',
              style:
                const TextStyle(fontSize: 160, fontWeight: FontWeight.w100),
            ),
            const Text(
              'Clicks',
              style: TextStyle(fontSize: 25),
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {
          setState(() {
            clickCounter++;
          });
        },
        child: const Icon(Icons.plus_one),
      ),
    );
  }
}

```