



Widget Flutter

Pertemuan ke 8: Statefull dan stateless, widget

Apa itu Widget?

Komponen dasar untuk membangun UI di Flutter. Semua yang dilihat di layar (tombol, teks, gambar) adalah widget.

Stateless Widget



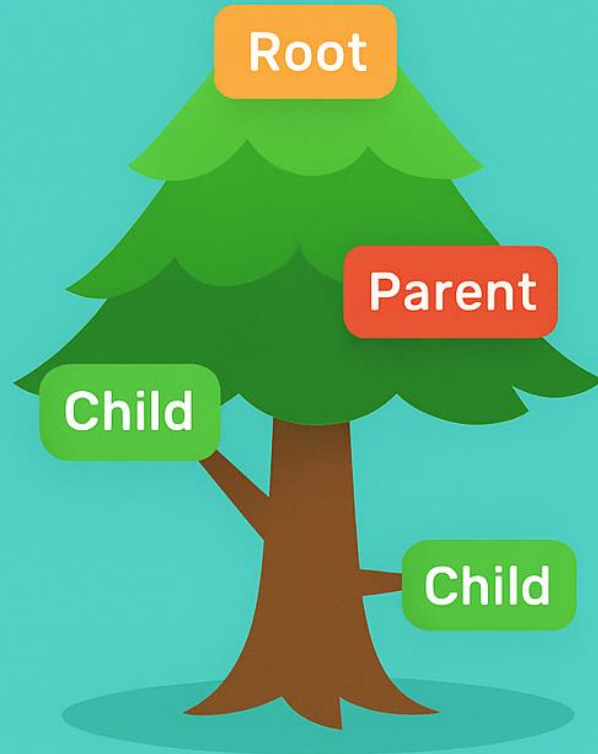
Dua jenis widget:

- **StatelessWidget:**
Tidak berubah seiring waktu
- **StatefulWidget:**
Bisa berubah berdasarkan interaksi atau data dinamis

Stateful Widget



WIDGET TREE DI FLUTTER



DEFINISI

- Struktur hierarki antar widget

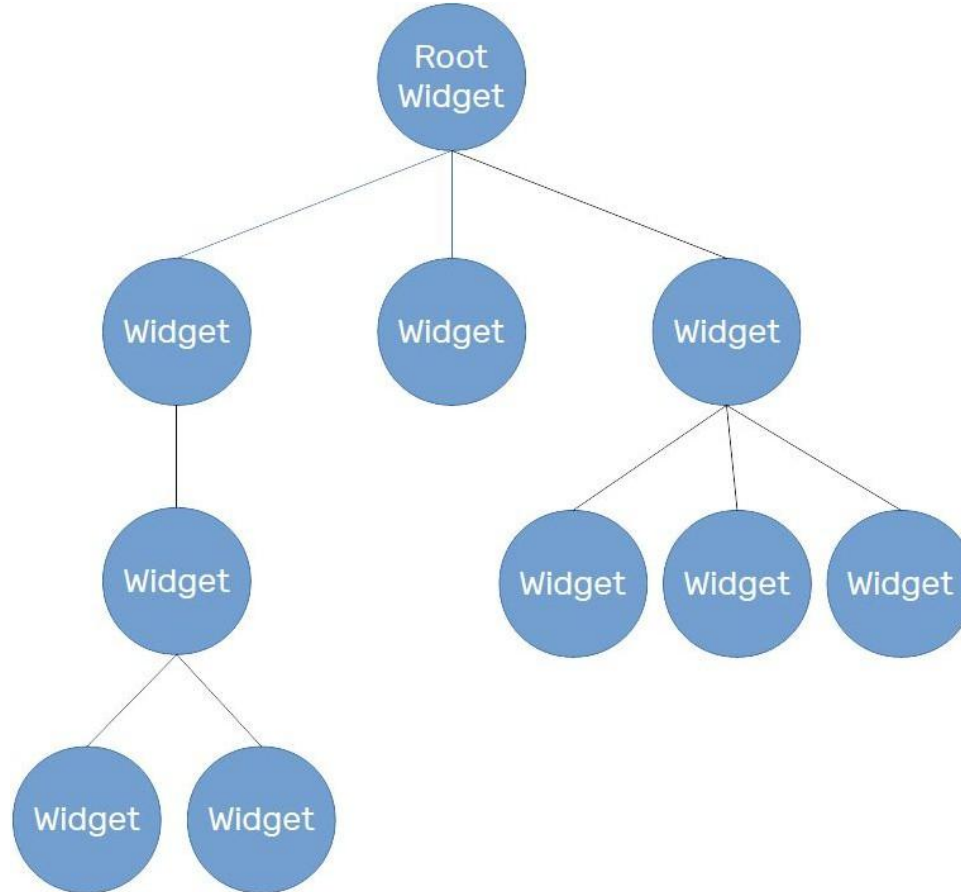
KONSEP

- Root: MaterialApp atau CupertinoApp
- Parent → Child

MANFAAT

- Memudahkan debugging UI
- Membuat kode lebih terstruktur
- Optimasi performa aplikasi

Widget Tree





Definisi

- Widget yang **tidak berubah** selama aplikasi berjalan
- Bersifat **statis** tampilan tetap dari awal hingga akhir

Ciri-ciri

- ✓ Tidak menyimpan **state**
- ✓ Dipanggil ulang hanya jika ada perubahan di **parent widget**
- ✓ Cocok untuk konten **tetap** (logo, teks statis, ikon)

Kapan Digunakan?

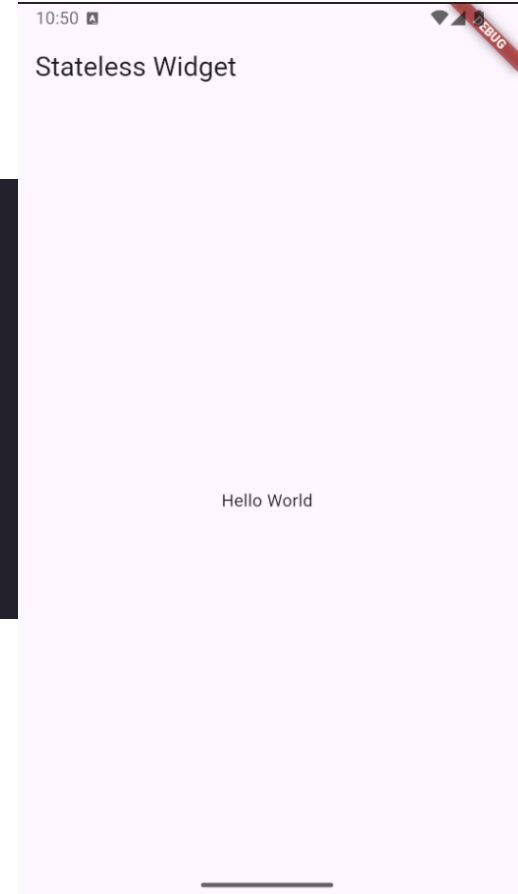
Saat UI tidak butuh update otomatis

Contoh Kode — StatelessWidget

```
1 class MyApp extends StatelessWidget {
2   const MyApp({super.key});
3
4   @override
5   Widget build(BuildContext context) {
6     return MaterialApp(
7       theme: ThemeData.fromSeed(seedColor:
8         Colors.deepPurple,),
9       home: Scaffold(
10        appBar: AppBar(title: const Text('Stateless
11          Widget')),
12        body: const Center(child: Text('Hello World')),
13      ),
14    );
15  }
```

Stateless Widget

```
class MyStatelessWidget extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return Text('Hello, Flutter!');  
  }  
}
```





● ● Contoh Kode

```
class MyCounterApp extends StatefulWidget {
  @override
  _MyCounterAppState createState() =
    ;
  class _MyCounterAppState
    extends State<MyCounterApp> {
    int _count = 0;
    // ...
  }
}
```

Definisi

- Widget yang **dapat berubah** selama aplikasi berjalan
- Menyimpan dan mengelola **state** (data dinamis) yang bisa diperbarui

⚡ Ciri-ciri

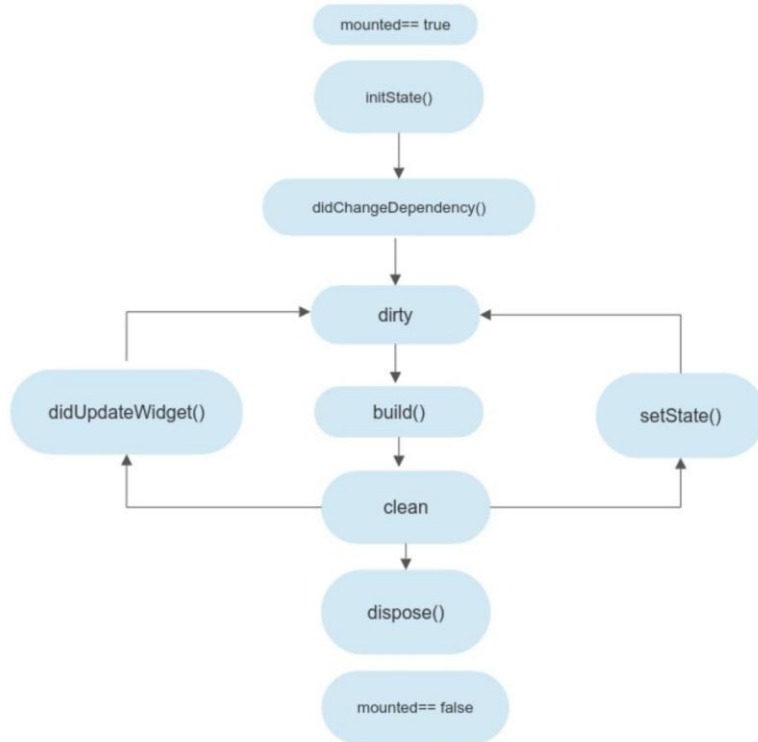
- ✓ Menyimpan **state** internal
- ✓ Tampilan dapat berubah se **interaksi pengguna** atau perubahan data
- ✓ Memiliki dua bagian: **StatefulWidget**

✓ Kapan Digunakan?

- Untuk data atau UI yang **berubah secara real-time**

Statefull Widget

State

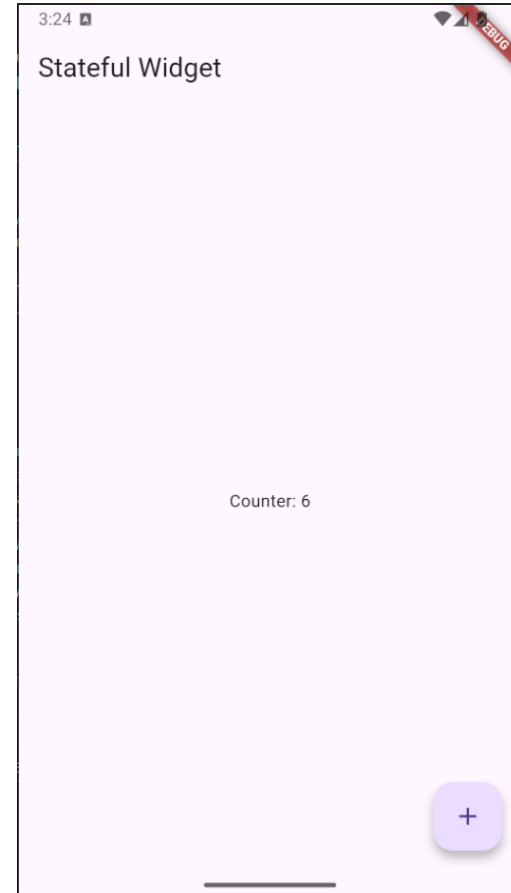


```

class _MyCounterAppState extends State<MyCounterApp> {
  int counter = 0;

  void _incrementCounter() {
    setState(() {
      counter++;
    });
  }

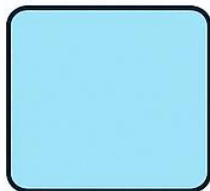
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Stateful Widget Demo',
      home: Scaffold(
        appBar: AppBar(title: const Text('Stateful Widget')),
        body: Center(child: Text('Counter: $counter')),
        floatingActionButton: FloatingActionButton(
          onPressed: _incrementCounter,
          child: const Icon(Icons.add),
        ), // FloatingActionButton
      ), // Scaffold
    ); // MaterialApp
  }
}
    
```



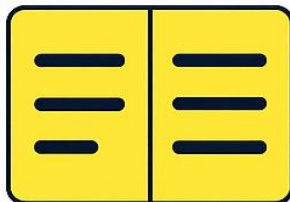
Text

Menampilkan
teks.



Container

Pembungkus
untuk styling
dan layout



**Row &
Column**

Menyusun widget
secara horizontal
dan vertikal



Image

Menampilkan
gambar



ElevatedButton

Tombol dengan
efek elevasi

Text Widget



```
Text(  
  'Halo, Flutter!',  
  style: TextStyle(fontSize: 24, color: Colors.blue),  
)
```

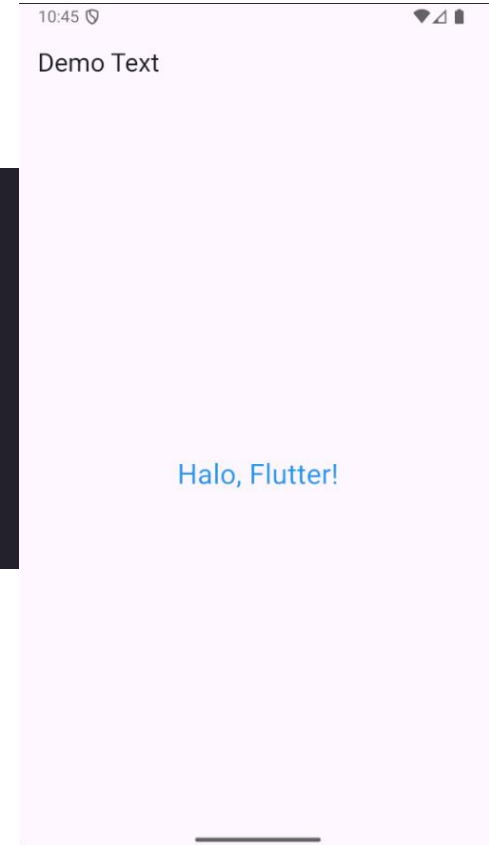


Image Widget

```
Image.asset(  
  'assets/image.png',  
  width: 200,  
  height: 200,  
)
```

10:49

Demo Gambar

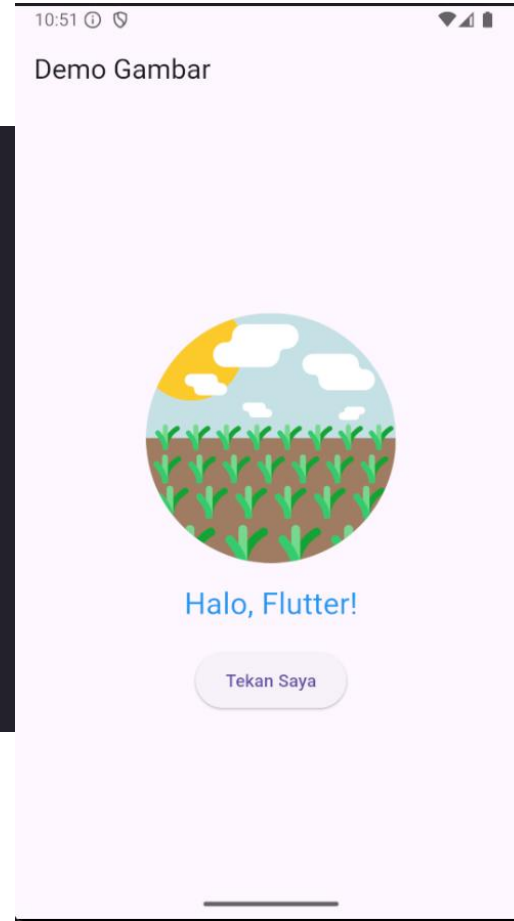


Halo, Flutter!



ElevatedButton Widget

```
ElevatedButton(  
  onPressed: () {  
    print('Tombol Ditekan');  
  },  
  child: Text('Tekan Saya'),  
)
```



LAYOUT DENGAN WIDGET



Row & Column

Untuk menyusun
widget di layar



Row

Menyusun widget
secara horizontal



Column

Menyusun
widget secara
vertikal

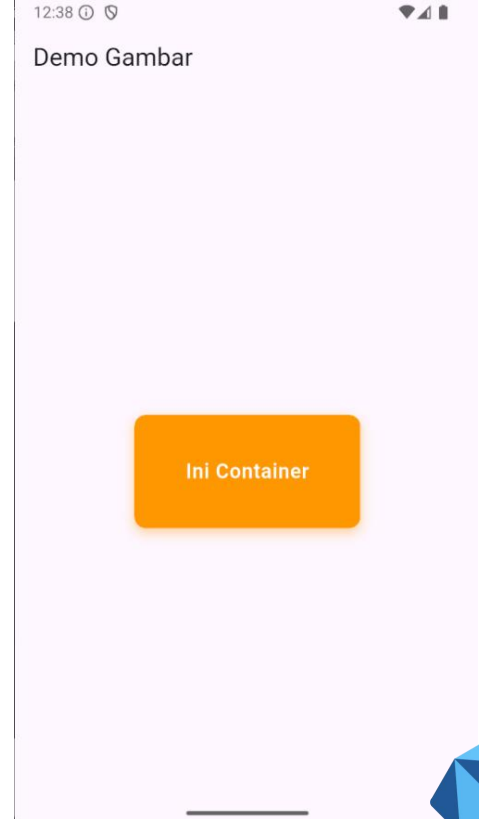


Container

Mengelola **padding**,
margin, dan
dekorasi widget

Container Widget

```
Container(  
  width: 200,  
  height: 100,  
  padding: EdgeInsets.all(10),  
  margin: EdgeInsets.all(20),  
  decoration: BoxDecoration(  
    color: Colors.orange,  
    borderRadius: BorderRadius.circular(10),  
  ),  
  child: Center(  
    child: Text(  
      'Ini Container',  
      style: TextStyle(color: Colors.white, fontSize: 18),  
    ),  
  ),  
)
```



Row & Column Widget

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

class MyRowColumnWidget extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Contoh Row & Column'),
      ),
      body: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Icon(Icons.star, color: Colors.red),
              SizedBox(width: 10),
              Text('Row 1, Item 1', style: TextStyle(fontSize: 18)),
            ],
          ),
          SizedBox(height: 20),
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
              Icon(Icons.star, color: Colors.green),
              SizedBox(width: 10),
              Text('Row 2, Item 1', style: TextStyle(fontSize: 18)),
            ],
          ),
        ],
      ),
    );
  }
}
```

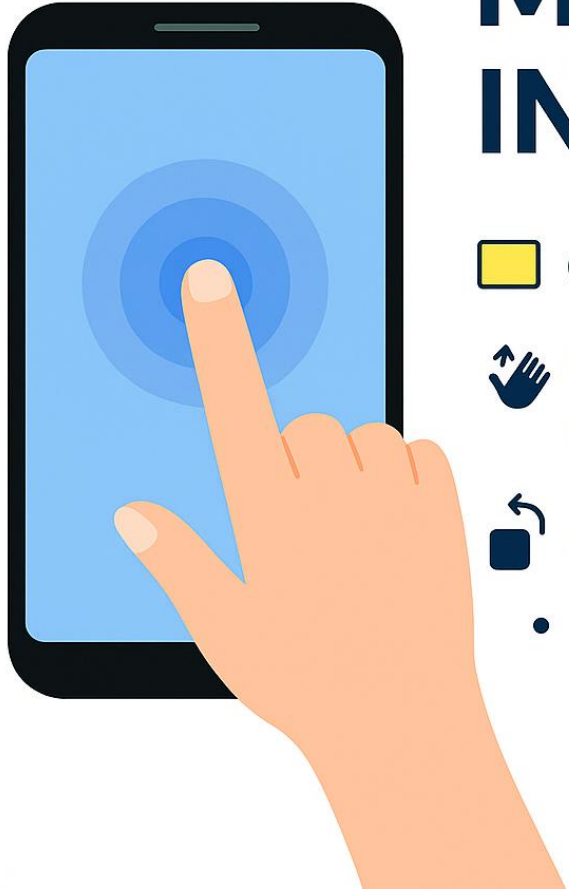
Contoh Row & Column

★ Row 1, Item 1

★ Row 2, Item 1



MENAMBAHKAN INTERAKTIVITAS



GestureDetector

 Menangkap gestur seperti tap, drag, dan swipe

InkWell

- Menampillkan efek visual saat widget di-tap

GestureDetector

```
GestureDetector(  
  onTap: () {  
    print('Widget di-tap!');  
  },  
  onTapDown: () {  
    print('Widget di-double tap!');  
  },  
  onLongPress: () {  
    print('Widget di-long press!');  
  },  
  child: Container(  
    width: 200,  
    height: 100,  
    color: Colors.blue,  
    child: Center(  
      child: Text(  
        'GestureDetector',  
        style: TextStyle(color: Colors.white, fontSize: 18),  
      ),  
    ),  
  ),  
)
```

12:53 ⓘ 🔒

GestureDetector Demo

GestureDetector



```
InkWell(  
  onTap: () {  
    print('InkWell di-tap!');  
  },  
  splashColor: Colors.red,  
  child: Container(  
    width: 200,  
    height: 100,  
    decoration: BoxDecoration(  
      color: Colors.green,  
      borderRadius: BorderRadius.circular(10),  
    ),  
    child: Center(  
      child: Text(  
        'InkWell',  
        style: TextStyle(color: Colors.white, fontSize: 18),  
      ),  
    ),  
  ),  
)
```

12:58 ⓘ 🔒



InkWell Demo

InkWell

