

Hello World App

Ketika pertama kali membuat project Flutter, kita akan diberikan aplikasi contoh yaitu aplikasi *counter*. *Source code* utama Flutter kita ada pada file `lib/main.dart` dan kodenya kurang lebih seperti berikut:

```
1. import 'package:flutter/material.dart';
2.
3. void main() {
4.   runApp(MyApp());
5. }
6.
7. class MyApp extends StatelessWidget {
8.   @override
9.   Widget build(BuildContext context) {
10.    return MaterialApp(
11.      title: 'Flutter Demo',
12.      theme: ThemeData(
13.        primarySwatch: Colors.blue,
14.        visualDensity: VisualDensity.adaptivePlatformDensity,
15.      ),
16.      home: MyHomePage(title: 'Flutter Demo Home Page'),
17.    );
18.  }
19. }
20.
21. class MyHomePage extends StatefulWidget {
22.   MyHomePage({Key key, this.title}) : super(key: key);
23.
24.   final String title;
25.
26.   @override
27.   _MyHomePageState createState() => _MyHomePageState();
28. }
29.
30. class _MyHomePageState extends State<MyHomePage> {
31.   int _counter = 0;
32.
33.   void _incrementCounter() {
34.     setState(() {
35.       _counter++;
36.     });
37.   }
38.
39.   @override
40.   Widget build(BuildContext context) {
41.     return Scaffold(
42.       appBar: AppBar(
43.         title: Text(widget.title),
44.       ),
```

```

45.         body: Center(
46.             child: Column(
47.                 mainAxisAlignment: MainAxisAlignment.center,
48.                 children: <Widget>[
49.                     Text(
50.                         'You have pushed the button this many times:',
51.                     ),
52.                     Text(
53.                         '$_counter',
54.                         style: Theme.of(context).textTheme.headline4,
55.                     ),
56.                 ],
57.             ),
58.         ),
59.         floatingActionButton: FloatingActionButton(
60.             onPressed: _incrementCounter,
61.             tooltip: 'Increment',
62.             child: Icon(Icons.add),
63.         ),
64.     );
65. }
66. }

```

Kode di atas merupakan kode *starter* yang di-generate sebagai contoh ketika Anda membuat project. Anda dapat mempelajari kode tersebut untuk mengetahui bagaimana sebuah aplikasi Flutter disusun.

Untuk saat ini kita tidak akan menggunakan kode tersebut. Jadi, hapus semua isi berkas main.dart tersebut dan tulis kode untuk aplikasi kita sendiri, yaitu aplikasi sederhana untuk menampilkan teks Hello world.

```

1. import 'package:flutter/material.dart';
2.
3. void main() {
4.   runApp(MyApp());
5. }
6.
7. class MyApp extends StatelessWidget {
8.   @override
9.   Widget build(BuildContext context) {
10.    return MaterialApp(
11.      title: 'Flutter Demo',
12.      theme: ThemeData(
13.        primarySwatch: Colors.blue,
14.      ),
15.      home: Scaffold(
16.        appBar: AppBar(
17.          title: Text('Hello, world!'),

```

```

18.     ),
19.     body: Center(
20.       child: Text('Hello, world!'),
21.     ),
22.   ),
23. );
24. }
25. }

```

Mari kita bahas satu per satu kodenya!

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

Import digunakan untuk memanggil fungsi-fungsi dari berkas Flutter yang lain. Pada kode di atas, kita meng-*import* kode-kode yang terdapat dalam *library* material bawaan Flutter. Library tersebut menyediakan widget yang termasuk dalam *material design*. Pastikan kode ini ada pada bagian atas sebelum kode lain dijalankan.

```
1. void main() => runApp(MyApp());
```

main() merupakan kode dasar dari project Flutter kita. Flutter akan menjalankan fungsi **main()** pertama kali, yang nantinya akan menjalankan **runApp()** dan memanggil **MyApp()**.

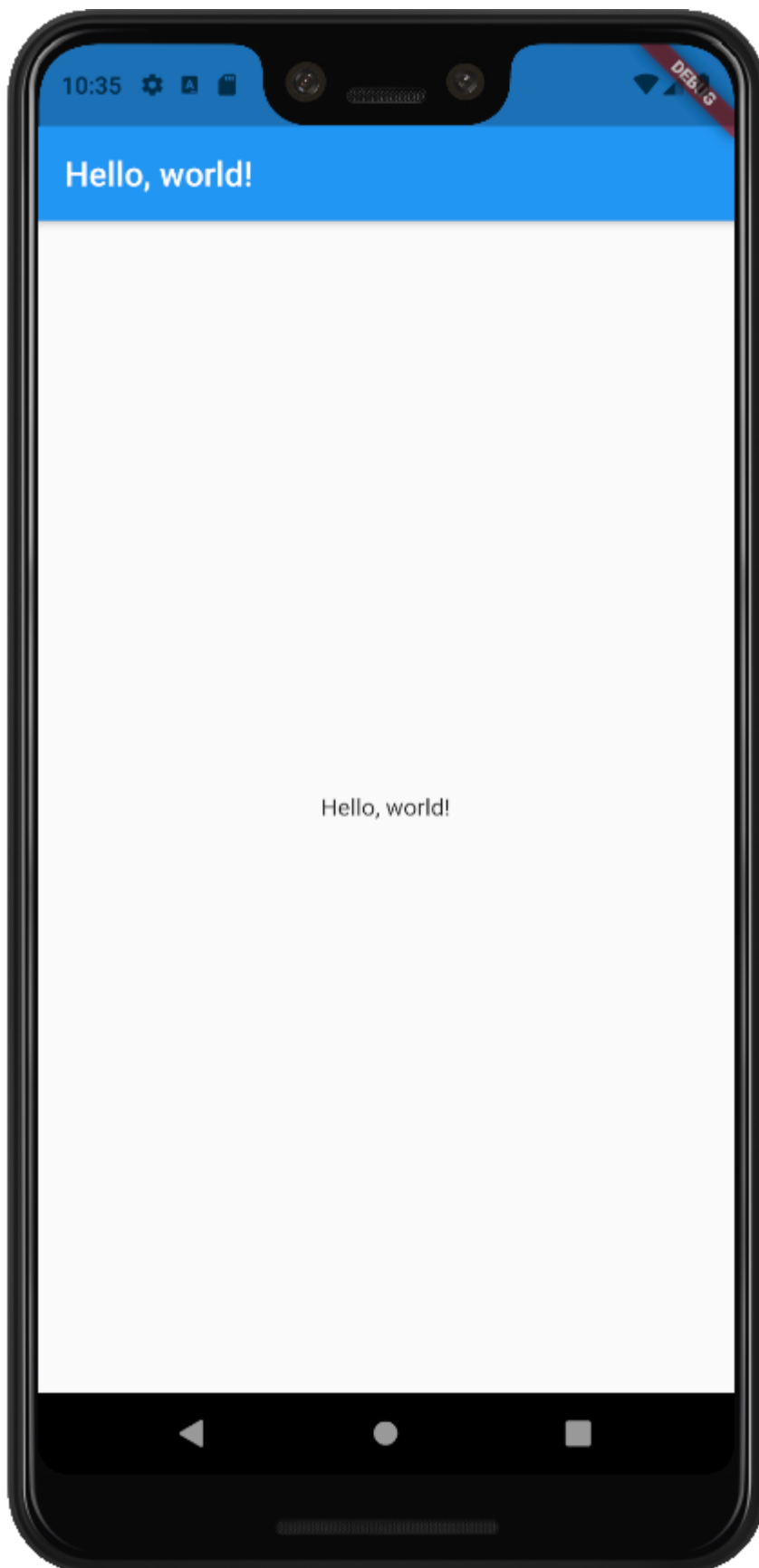
```

1. class MyApp extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return MaterialApp(
5.       title: 'Flutter Demo',
6.       theme: ThemeData(
7.         primarySwatch: Colors.blue,
8.       ),
9.       home: Scaffold(
10.        appBar: AppBar(
11.          title: Text('Hello, world!'),
12.        ),
13.        body: Center(
14.          child: Text('Hello, world!'),
15.        ),
16.      ),
17.    );
18.  }
19. }

```

MyApp di sini merupakan sebuah *class* yang menampilkan komponen Flutter ke layar atau dikenal dengan Widgets.

Jika aplikasi dijalankan maka akan seperti berikut:



Setiap komponen di dalam Flutter terdiri dari widget. Bahkan aplikasi Flutter itu sendiri merupakan sebuah widget. Pada contoh kode di atas kelas MyApp yang merupakan sebuah widget mengembalikan atau menampilkan widget MaterialApp. MaterialApp ini adalah widget bawaan Flutter yang akan menjadi fondasi dari aplikasi Anda. Ia umum digunakan supaya aplikasi bisa menerapkan material design. Widget ini mengatur beberapa hal, termasuk tema aplikasi, tampilan utama aplikasi, rute untuk navigasi antar halaman, dan lain-lain.

Selanjutnya Scaffold memungkinkan kita mendesain struktur layout dasar yang sesuai dengan material design. Dengan Scaffold Anda dapat mengatur App Bar, Floating Action Button, Drawer, dan lain-lain.

Ibarat rumah, MaterialApp ini adalah sebagai tanah yang umumnya digunakan hanya sekali saja, sedangkan Scaffold ini adalah sebagai tembok atau bangunannya.

Terakhir, seharusnya sudah cukup jelas kita menggunakan widget Text untuk menampilkan teks. Pada contoh diatas widget Text dibungkus dengan widget Center yang berfungsi supaya widget di dalamnya (*child*) tampil pada posisi tengah.

Tips: *Documentation is your best friend.*

Flutter memiliki banyak sekali widget dan tentunya akan menjadi tugas yang berat untuk menghafalkan semuanya. Namun tenang, Flutter dilengkapi dengan dokumentasi yang cukup lengkap sebagai panduan kita dalam mempelajari dan mengembangkan Flutter. Misalnya, penjelasan lebih lengkap tentang widget yang kita gunakan di atas dapat dilihat pada tautan berikut:

- <https://api.flutter.dev/flutter/material/MaterialApp-class.html>
- <https://api.flutter.dev/flutter/material/Scaffold-class.html>
- <https://api.flutter.dev/flutter/widgets/Center-class.html>
- <https://api.flutter.dev/flutter/widgets/Text-class.html>

