

PRAKTIKUM 4 dan 5
Widget - Widget Umum
Scaffold
(AppBar, Body, FloatingActionButton)
Mobile Application



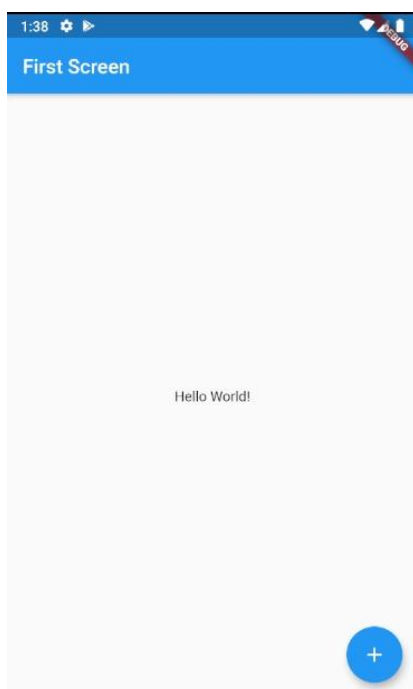
Basworo Ardi Pramono
Fakultas Teknologi Informasi dan Komunikasi
Universitas Semarang
2021

Setelah sebelumnya kita belajar mengenai StatelessWidget dan StatefulWidget, kali ini kita akan belajar widget-widget yang umum digunakan dalam pengembangan aplikasi Flutter. Dengan widget-widget tersebut, kita akan belajar bagaimana membuat sebuah tampilan (material design), layout sederhana, button, serta input control.

Untuk mencoba widget-widget umum tersebut, buatlah project baru lalu pada file lib/main.dart buatlah kode seperti yang ada pada pembahasan Aplikasi Hello World.

Scaffold

Scaffold merupakan sebuah widget yang digunakan untuk membuat tampilan dasar material design pada aplikasi Flutter, yang dapat disebut juga dasar sebuah halaman pada aplikasi Flutter. Tampilan dasar tersebut seperti berikut:

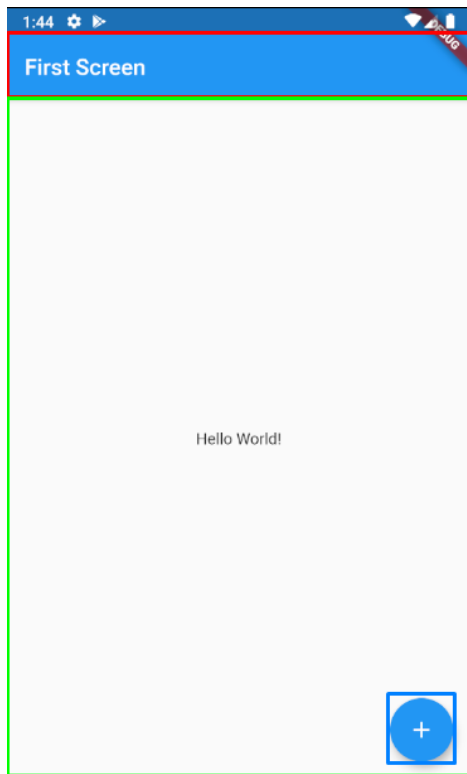


Tampilan di atas merupakan implementasi dari Scaffold.

Scaffold di atas memiliki 3 bagian yaitu

1. **AppBar,**
2. **Body, dan**
3. **FloatingActionButton.**

Ketiga bagian tersebut diilustrasikan seperti berikut:



Pada gambar di atas

1. kotak berwarna merah merupakan AppBar;
2. kotak berwarna hijau merupakan body; dan
3. kotak berwarna biru merupakan FloatingActionButton.

Untuk membuat sebuah Scaffold kita hanya cukup memanggil class **Scaffold** seperti berikut:

1. `class FirstScreen extends StatelessWidget {`
2. `@override`
3. `Widget build(BuildContext context) {`
4. `return Scaffold();`
5. `}`
6. `}`

```

117
118   class FirstScreen extends StatelessWidget {
119       @override
120       Widget build(BuildContext context) {
121           return Scaffold();
122       }
123   }
124

```

Pada kode di atas kita membuat sebuah StatelessWidget bernama FirstScreen, yang merupakan widget tampilan kita. Kemudian di dalam method build kita panggil Scaffold.

Jangan lupa untuk memanggil FirstScreen pada Widget MyApp seperti berikut:

```

1   import 'package:flutter/material.dart';
2
3   void main() => runApp(MyApp());
4
5   class MyApp extends StatelessWidget {
6       @override
7       Widget build(BuildContext context) {
8           return MaterialApp(
9               title: 'Flutter Demooo',
10              theme: ThemeData(
11                  primarySwatch: Colors.blue,
12              ), // ThemeData
13              home: FirstScreen(), // Panggil FirstScreen di sini
14          ); // MaterialApp
15      }
16  }
17
18
19  class FirstScreen extends StatelessWidget {
20      @override
21      Widget build(BuildContext context) {
22          return Scaffold();
23      }
24  }
25

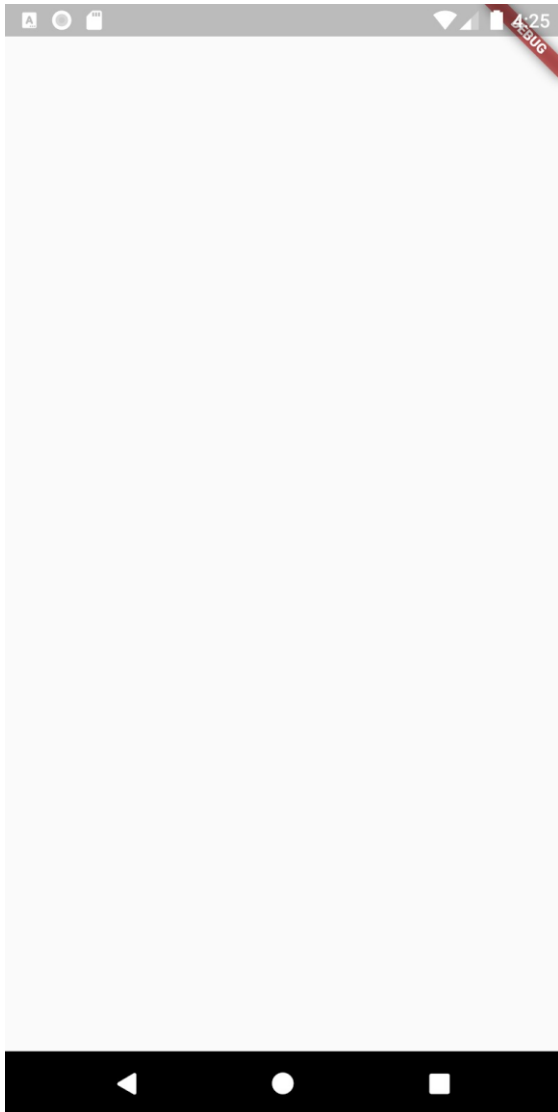
```

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

- 2.

```
3. void main() => runApp(MyApp());
4.
5. class MyApp extends StatelessWidget {
6.   // This widget is the root of your application.
7.   @override
8.   Widget build(BuildContext context) {
9.     return MaterialApp(
10.       title: 'Flutter Demo',
11.       theme: ThemeData(
12.         primarySwatch: Colors.blue,
13.       ),
14.       home: FirstScreen(), // Panggil FirstScreen di sini
15.
16.     );
17.   }
18. }
19.
20. class FirstScreen extends StatelessWidget {
21.   @override
22.   Widget build(BuildContext context) {
23.     return Scaffold();
24.   }
25. }
```

Ketika kita menjalankan aplikasi Flutter, pada layar akan hanya muncul tampilan berwarna putih.



AppBar

Setelah kita membuat Scaffold pertama kita pada Widget FirstScreen, sekarang kita akan menambahkan AppBar pada Scaffold. Seperti yang kita tahu AppBar merupakan Header (bagian paling atas) aplikasi atau biasa dikenal dengan toolbar. Pada AppBar umumnya terdapat judul dan ActionButton.

Berikut adalah cara menambahkan AppBar pada Scaffold:

1. `class FirstScreen extends StatelessWidget {`
2. `@override`
3. `Widget build(BuildContext context) {`

```

4.   return Scaffold(
5.     appBar: AppBar(
6.       title: Text('First Screen'),
7.     ),
8.   );
9. }
10.}

```

Pada kode di atas kita menambahkan parameter **appBar** pada **Scaffold** dan menambahkan title pada AppBar tersebut. Title di sini tidak hanya spesifik Text saja, melainkan juga dapat diisi dengan widget lainnya seperti

TextField untuk kolom pencarian atau yang lainnya. Setelah menambahkan kode di atas, coba refresh atau hot reload aplikasi Flutter Anda. Selain menambahkan title kita dapat menambahkan widget-widget actions seperti pada kode berikut:

```

1. class FirstScreen extends StatelessWidget {
2.   @override
3.   Widget build(BuildContext context) {
4.     return Scaffold(
5.       appBar: AppBar(
6.         title: Text('First Screen'),
7.         actions: <Widget>[
8.           IconButton(
9.             icon: Icon(
10.              Icons.search,
11.              color: Colors.white,
12.            ),
13.            onPressed: () {},
14.          ),
15.        ],
16.      ),

```

```
17. );
```

```
18. }
```

```
19. }
```

Pada kode di atas kita menambahkan Icon search pada bagian kanan AppBar. Lalu kita juga dapat menambahkan action pada bagian kiri AppBar misalnya untuk tombol yang menampilkan menu (drawer), seperti pada kode berikut:

```
1. class FirstScreen extends StatelessWidget {
```

```
2.   @override
```

```
3.   Widget build(BuildContext context) {
```

```
4.     return Scaffold(  
5.       appBar: AppBar(  
6.         leading: IconButton(  
7.           icon: Icon(  
8.             Icons.menu,  
9.             color: Colors.white,  
10.          ),  
11.          onPressed: () {},  
12.        ),  
13.        title: Text('First Screen'),  
14.        actions: [  
15.          IconButton(  
16.            icon: Icon(  
17.              Icons.search,  
18.              color: Colors.white,  
19.            ),  
20.            onPressed: () {},  
21.          ),
```



```
22.     ],
```

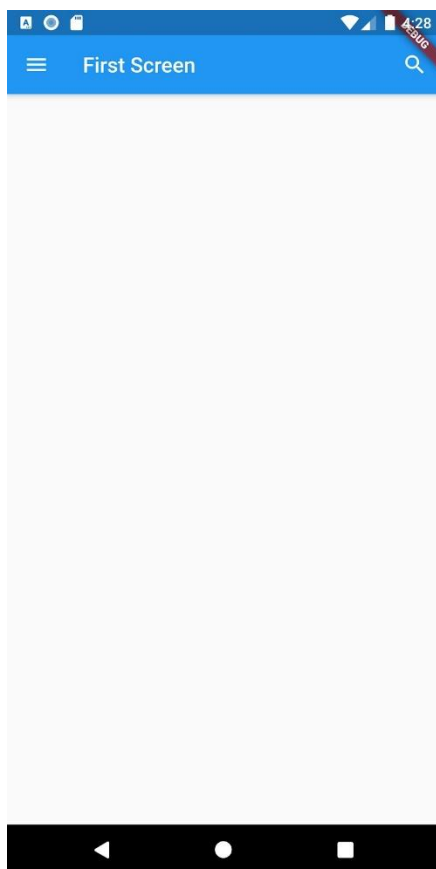
```
23.     ),
```

```
24.   );
```

```
25. }
```

```
26. }
```

Tidak seperti pada actions, leading hanya dapat menampung satu widget saja. Secara default, leading akan berisi tombol untuk kembali ke halaman sebelumnya (jika tersedia), atau tombol untuk menu drawer (jika kita mengatur untuk drawer pada Scaffold tersebut). Untuk melihat hasilnya lakukan refresh atau hot reload pada aplikasi Flutter Anda.

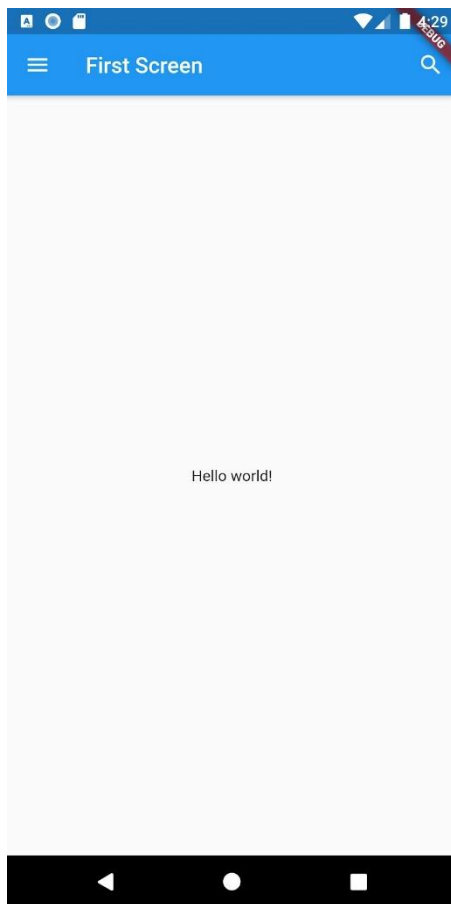


Body

Setelah menambahkan AppBar kita akan menambahkan body. Seperti pada ilustrasi sebelumnya, body merupakan bagian utama dari Scaffold dan kita akan banyak menuliskan kode pada bagian body ini. Untuk implementasi body kita akan menambahkan parameter body pada Scaffold seperti berikut:

```
1. class FirstScreen extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Scaffold(  
5.       appBar: AppBar(  
6.         leading: IconButton(  
7.           icon: Icon(Icons.menu, color: Colors.white,),  
8.           onPressed: () {},  
9.         ),  
10.      title: Text('First Screen'),  
11.      actions: <Widget>[  
12.        IconButton(  
13.          icon: Icon(Icons.search, color: Colors.white,),  
14.          onPressed: () {},  
15.        )  
16.      ],  
17.    ),  
18.    body: Center(  
19.      child: Text('Hello world!'),  
20.    ),  
21.  );  
22. }  
23. }
```

Pada kode di atas kita telah menambahkan body yang di dalamnya kita memanggil widget Center yang akan menampilkan Text "Hello World!".



FloatingActionButton

Selanjutnya, kita akan menambahkan sebuah tombol bulat pada bagian kanan bawah seperti ilustrasi sebelumnya yaitu FloatingActionButton. FloatingActionButton ini merupakan bagian dari Scaffold yang digunakan untuk menampilkan sebuah tombol aksi yang posisinya floating (melayang dan posisinya tetap). Untuk menggunakan FloatingActionButton tambahkan kode Anda seperti berikut:

```
1. class FirstScreen extends StatelessWidget {  
2.   @override  
3.   Widget build(BuildContext context) {  
4.     return Scaffold(  
5.       appBar: AppBar(  
6.         leading: IconButton(  
7.           icon: Icon(Icons.menu, color: Colors.white,),  
8.           onPressed: () {},  
9.         ),  
10.        title: Text('First Screen'),  
11.        actions: <Widget>[  
12.          IconButton(  
13.            icon: Icon(Icons.search, color: Colors.white,),  
14.            onPressed: () {},  
15.          )  
16.        ],  
17.      ),  
18.      body: Center(  
19.        child: Text('Hello world!'),  
20.      ),  
21.      floatingActionButton: FloatingActionButton(  
22.        child: Icon(Icons.add),  
23.      ),  
24.    );  
25.  }  
26. }
```

Hasil Akhir

Setelah kita menambahkan **AppBar**, **body**, dan **FloatingActionButton** maka hasil akhirnya akan seperti berikut:

